LAMP-TR-056
CAR-TR-951
CS-TR-4174

MDA 9049-6C-1250
August 2000

Learning Algorithms for Audio and Video Processing—Independent Component Analysis and Support Vector Machine Based Approaches

Yuan Qi

Center for Automation Research
University of Maryland
College Park, MD 20742-3275

Abstract

In this thesis, we propose two new machine learning schemes, a Subband-based Independent Component Analysis scheme and a hybrid Independent Component Analysis/Support Vector Machine scheme, and apply them to the problems of blind acoustic signal separation and face detection.

Based on a linear model, classical Independent Component Analysis (ICA) provides a method of representing data as independent components. In contrast to Principal Component Analysis (PCA), which decorrelates the data based on its covariance matrix, ICA uses higher-order statistics of the data to minimize the dependence between the components of the system output. An important application of ICA is blind source separation. However, classical ICA algorithms do not work well for separation in the presence of noise or when performed on-line. Inspired by the psychoacoustic discovery that humans perceive and process acoustic signals in different frequency bands independently, we propose a new algorithm, subband-based ICA, that integrates ICA with time-frequency analysis to separate mixed signals. In subband-based ICA, the separations are performed in parallel in several frequency bands. Wavelet decomposition and best basis selection in wavelet/DCT packets can be incorporated into this algorithm. Subband-based ICA is computationally fast, robust to noise, and works well in an on-line version when other ICA algorithms fail. The virtually increased signal-to-noise ratio in those frequency bands where the separations are actually performed, and the fact that subband signals, i.e., wavelet coefficients, are more peaky and heavy-tailed distributed than the original signals, both contribute to the success of subband-based ICA. Experimental results on separating noisy speech mixtures and musical signal mixtures demonstrate its effectiveness.

This research was funded in part by the Department of Defense and the Army Research Laboratory under Contract MDA 9049-6C-1250. Thanks to Sara Larson for formatting this report.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
AUG 2000		**00-08-2000 to 00-08-2000**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Learning Algorithms for Audio and Video Processing - Independent Component Analysis and Support Vector Machine Based Approaches	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Language and Media Processing Laboratory,Institute for Advanced Computer Studies,University of Maryland,College Park,MD,20742-3275**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**54**	

In addition to separating mixed signals, ICA can also be used as a feature extractor. As argued by many researchers in the neural research area, a principle of sensory information processing in the brain is redundancy reduction. The ICA representation of the data follows this principle. Also, from a signal processing viewpoint, ICA provides a nice way to cluster independent signals and hence leads to a better representation of signals than PCA.

Motivated by the feature extraction capability of ICA, we propose a new hybrid unsupervised/supervised learning scheme that integrates Independent Component Analysis with the Support Vector Machine (SVM) approach and apply this new learning scheme to the face detection problem. SVM is a new powerful machine learning algorithm which is rooted in statistical learning theory. As an approximate implementation of the Structural Risk Minimization (SRM) Principle proposed in statistical learning theory, SVM tends to have good generalization performance. One common characteristic shared by ICA and SVM is sparsity. The ICA output is sparse, and the support vectors whose linear combination comprises the trained SVM are also sparse. Thus integrating ICA with SVM yields a new hybrid hierarchical sparse learning scheme.

Specifically, for the face detection problem we use ICA in two different ways to extract low-level features from a window sliding over an image, and then apply SVM at a high level to classify the extracted ICA features as a face or not. Experimental results show that using the first method to extract ICA features and applying SVM for classification effectively improves the detection system performance, compared with applying SVM directly to the original image data.

Finally as a general learning scheme, hybrid ICA/SVM can be applied to other pattern recognition problems as well as to face detection.

LAMP-TR-056
CAR-TR-951
CS-TR-4174

MDA-9049-6C-1250
August 2000

Learning Algorithms for Audio and Video Processing—Independent Component Analysis and Support Vector Machine Based Approaches

Yuan Qi

Learning Algorithms for Audio and Video Processing—Independent Component Analysis and Support Vector Machine Based Approaches

Yuan Qi

Center for Automation Research
University of Maryland
College Park, MD 20742-3275

Abstract

In this thesis, we propose two new machine learning schemes, a Subband-based Independent Component Analysis scheme and a hybrid Independent Component Analysis/Support Vector Machine scheme, and apply them to the problems of blind acoustic signal separation and face detection.

Based on a linear model, classical Independent Component Analysis (ICA) provides a method of representing data as independent components. In contrast to Principal Component Analysis (PCA), which decorrelates the data based on its covariance matrix, ICA uses higher-order statistics of the data to minimize the dependence between the components of the system output. An important application of ICA is blind source separation. However, classical ICA algorithms do not work well for separation in the presence of noise or when performed on-line. Inspired by the psychoacoustic discovery that humans perceive and process acoustic signals in different frequency bands independently, we propose a new algorithm, subband-based ICA, that integrates ICA with time-frequency analysis to separate mixed signals. In subband-based ICA, the separations are performed in parallel in several frequency bands. Wavelet decomposition and best basis selection in wavelet/DCT packets can be incorporated into this algorithm. Subband-based ICA is computationally fast, robust to noise, and works well in an on-line version when other ICA algorithms fail. The virtually increased signal-to-noise ratio in those frequency bands where the separations are actually performed, and the fact that subband signals, i.e., wavelet coefficients, are more peaky and heavy-tailed distributed than the original signals, both contribute to the success of subband-based ICA. Experimental results on separating noisy speech mixtures and musical signal mixtures demonstrate its effectiveness.

In addition to separating mixed signals, ICA can also be used as a feature extractor. As argued by many researchers in the neural research area, a principle of sensory information

processing in the brain is redundancy reduction. The ICA representation of the data follows this principle. Also, from a signal processing viewpoint, ICA provides a nice way to cluster independent signals and hence leads to a better representation of signals than PCA.

Motivated by the feature extraction capability of ICA, we propose a new hybrid unsupervised/supervised learning scheme that integrates Independent Component Analysis with the Support Vector Machine (SVM) approach and apply this new learning scheme to the face detection problem. SVM is a new powerful machine learning algorithm which is rooted in statistical learning theory. As an approximate implementation of the Structural Risk Minimization (SRM) Principle proposed in statistical learning theory, SVM tends to have good generalization performance. One common characteristic shared by ICA and SVM is sparsity. The ICA output is sparse, and the support vectors whose linear combination comprises the trained SVM are also sparse. Thus integrating ICA with SVM yields a new hybrid hierarchical sparse learning scheme.

Specifically, for the face detection problem we use ICA in two different ways to extract low-level features from a window sliding over an image, and then apply SVM at a high level to classify the extracted ICA features as a face or not. Experimental results show that using the first method to extract ICA features and applying SVM for classification effectively improves the detection system performance, compared with applying SVM directly to the original image data.

Finally as a general learning scheme, hybrid ICA/SVM can be applied to other pattern recognition problems as well as to face detection.

Chapter 1

Introduction

In this thesis, we propose two new machine learning schemes, a Subband-based Independent Component Analysis scheme and a hybrid Independent Component Analysis/Support Vector Machine scheme, and apply them in the problems of blind acoustic signal separation and face detection. This introduction provides brief summaries of these two learning schemes and an outline of the dissertation. In Section 1.1 we give a short background review of Independent Component Analysis and Support Vector Machines. The motivations and concise descriptions of our two new learning schemes are described in Section 1.2, and the organization of the dissertation is outlined in Section 1.3.

1.1 Background Review

1.1.1 Independent Component Analysis

A common problem in statistics, signal processing, and neural network research is how to design an appropriate representation for multivariate data. Based on a linear model, Independent Component Analysis offers a method of representing the data as independent components. In contrast to Principal Component Analysis, which decorrelates the data based on its covariance matrix, ICA uses higher-order statistics of the data to minimize the dependence between the components of the representation. Such a representation seems to capture the essential structure of the data in many problems. As a result, ICA is being used in an increasing number of applications, such as speech enhancement and recognition, telecommunication, biomedical signal analysis, and image denoising and recognition [3, 12, 39, 8, 42, 36, 21]. In these applications, the problems to which ICA is applied include blind source separation, blind deconvolution, and feature extraction.

In the blind source separation problem, ICA is applied to recover independent unknown sources given only sensor observations that are unknown linear mixtures of the unobserved sources and noise. ICA has been successfully applied to separate acoustic signals, electroencephalographic (EEG) signals, and magnetoencephalographic (MEG) signals. Also, ICA has been used in the blind equalization and Code Division Multiple Access (CDMA) system in communications. For the blind deconvolution problem, if we transform the data to the frequency domain, the problem becomes the same as the blind separation problem, so that it can be tackled by ICA too.

In the feature extraction problem, ICA aims to find an independent basis or representation coefficients for the data. In [7, 6, 5, 20], Barlow et al. argued that a principle of sensory information processing in the brain is redundancy reduction. The ICA representation of the data follows this principle. In [9] Bell and Sejnowski point out that the independent components

of natural scenes are edge filters. In [48], Olshausen and Field show, under the noise-free assumption, an equivalence between an ICA algorithm and sparse coding, another method of implementing the redundancy reduction principle. In [63], Hateren et al. report a detailed comparison between ICA features and the properties of simple cells in the macaque primary visual cortex, and find good matches to most of the parameters. Besides these discoveries of psychological and neural research, from the signal processing viewpoint, ICA provides a nice way of clustering independent signals and hence leads to a better representation of signals than classical Principal Component Analysis. This also justifies the use of ICA for feature extraction. ICA feature extraction has been applied to face recognition and image denoising and satisfying results have been obtained.

1.1.2 Support Vector Machines and Statistical Learning Theory

The Support Vector Machine (SVM) is a powerful machine learning algorithm, which is rooted in statistical learning theory. According to the *Structural Risk Minimization (SRM)* Principle in statistical learning theory [65], the error rate of a learning machine on test data is bounded by the sum of the training error rate and a term that depends on the *Vapnik-Chervonenkis (VC) dimension* and indicates the complexity of the model. By first nonlinearly mapping the input data into a high-dimensional feature space, and then constructing a hyperplane as the decision surface in that space which leaves the maximal margin between positive and negative examples, SVM approximately implements the SRM Principle. Thus the training error rate and the model complexity can be minimized at the same time by SVM. Therefore, in theory, SVM tends to have good generalization performances. Many applications have also demonstrated the good generalization performance of SVM, including isolated handwritten digit recognition [58], object recognition [10], speaker identification [57], and face detection [49].

In addition to good generalization performance, SVM has many other nice properties:

- By reformulating the primary quadratic programming (QP) problem encountered in SVM training into its dual problem and using a suitable inner-product kernel, SVM controls the model complexity independently of the dimensionality of the feature space. Actually, infinite feature spaces are allowed in SVM.

- Moreover, the convex cost function in the QP problem guarantees that SVM will find a globally optimal solution, while many other learning algorithms suffer from falling into local extrema.

- By solving the QP problem during the training phase, SVM automatically tunes all the parameters in a learning scheme.

- The support vectors, whose linear combination comprises the trained SVM, are usually *sparse*. By reformulating SVM in the framework of regularization theory, Girosi [29] shows an equivalence between SVMs and a Sparse Approximation (SA) scheme that resembles the Basis Pursuit De-Noising algorithm [14]. This reveals the relationship between SVM and other known techniques.

1.2 Motivation and Contributions

Motivated by discoveries in mammalian acoustic and visual systems, we propose two new learning schemes for acoustic and visual signal processing, which are briefly described in the following sections.

1.2.1 Integration of ICA and Time-Frequency Analysis

Though classical ICA algorithms have been applied to address the problem of blind source separation, they do not work well in the presence of noise or when performed on-line. Inspired by the psychoacoustic discovery that humans perceive and process acoustic signals in different frequency bands independently [1, 43], we propose a new algorithm, subband-based ICA, that integrates ICA with time-frequency analysis to separate mixed signals. In subband-based ICA, the separations are performed in parallel in several frequency bands. Wavelet decomposition and best basis selection in wavelet/DCT packets can be incorporated into this algorithm. Subband based ICA is computationally fast, robust to noise, and works well in an on-line version when other ICA algorithms fail. The virtually increased signal-to-noise ratio in those frequency bands, the fact that subband signals, i.e., wavelet coefficients, are more peaky and heavy-tailed distributed than the original signals, and the adaptation to the properties of the signal and noise by the incorporation of best basis selection algorithm, all contribute to the success of subband-based ICA. Experimental results on separating noisy speech mixtures and musical signal mixtures demonstrate its effectiveness.

1.2.2 Face Detection Based on the Hybrid ICA/SVM Learning Scheme

Motivated by the feature extraction capability of ICA as mentioned in Section 1.1.1, we propose a new hybrid unsupervised/supervised learning scheme that integrates Independent Component Analysis with the Support Vector Machine and we apply this new learning scheme to the face detection problem. Specifically, we use ICA in two different ways to extract low-level features from a window sliding over an image, and then apply SVM at a high level to decide whether the extracted ICA features represent a face. Experimental results show that using the first method of extracting ICA features effectively improves detection system performance, compared with applying SVM directly to the original image data.

An interesting comment about the hybrid learning scheme is that ICA and SVM share a common characteristic, sparsity. The ICA output is sparse, and the support vectors in SVM are also sparse. Hence the hybrid learning scheme has a hierarchical sparse architecture.

Furthermore, as a general learning scheme, hybrid ICA/SVM can be applied to pattern recognition problems other than face detection.

1.3 Thesis Outline

The rest of the thesis is organized as follows. In Chapter 2, we introduce the classical ICA algorithm, propose the new subband-based ICA, and apply the new algorithm to separating mixed acoustic signals. In Chapter 3, we present a review and discussion of SVM. Finally, in Chapter 4, we propose the hybrid ICA/SVM learning scheme, and apply it to the face detection problem.

Chapter 2

Subband-based Independent Component Analysis

2.1 Introduction

Independent Component Analysis (ICA) can recover independent sources given only sensor observations that are unknown linear mixtures of the unobserved source signals and noise. In contrast to Principal Component Analysis, which decorrelates signals based on the covariance matrix, ICA uses higher-order statistics of the signals to find independent components. ICA has many applications in speech enhancement and recognition, telecommunication, biomedical signal analysis, and image denoising and recognition [3, 12, 39, 8, 42, 36]. However, classical ICA algorithms do not work well on-line or in the presence of noise. Inspired by the psychoacoustic discoveries connecting auditory perception and wavelet theory, a new ICA algorithm, subband-based ICA, is proposed to separate independent signals. Experimental results on separating mixed acoustic signals demonstrate its robustness to noise and its high efficiency when performed on-line.

2.2 Classical ICA System Model and Learning Rule

While several nonlinear ICA algorithms have been proposed [37, 40], most of the contributions to the ICA literature are based on the linear input mixture model, which is defined as

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{b}(t),$$

where $\mathbf{s}(t) = [s_1(t), s_2(t), \ldots, s_n(t)]^T$ is an unknown source signal vector at discrete time t, $\mathbf{x}(t) = [x_1(t), x_2(t), \ldots, x_n(t)]^T$ is the observation signal vector, $\mathbf{A}$ is a full-rank $n \times n$ mixing matrix, and $\mathbf{b}(t)$ is noise. The components of the vector $\mathbf{s}(t)$, i.e., $s_1(t), s_2(t), \ldots, s_n(t)$, come from n independent sources. Unlike factor analysis addressed by an EM algorithm [28], which assumes that $\mathbf{b}(t)$ is normally distributed with a diagonal covariance matrix and $\mathbf{s}(t)$ is also normally distributed, ICA algorithms are derived on the assumption of noise-free measurements. In practice, many ICA algorithms do not work well on noisy mixtures.

Given the mixture model, the aim of ICA is to recover the original source signal $\mathbf{s}(t)$. To this end, the following simple separation model is used, corresponding to the above linear mixture model:

$$\mathbf{y}(t) = \mathbf{W}\mathbf{x}(t),$$

where $\mathbf{y}(t) = [y_1(t), y_2(t), \ldots, y_n(t)]^T$ is an estimate of $\mathbf{s}(t)$ and $\mathbf{W}$ is the unmixing matrix, i.e., an estimate of the inverse of $\mathbf{A}$.

To obtain the learning rule for the unmixing matrix $\mathbf{W}$, we use the natural gradient [4] to minimize the Kullback-Leibler divergence between the source signal vector $\mathbf{s}$ and its estimate

$\mathbf{y}$, i.e.,

$$D(f_{\mathbf{y}} \parallel f_{\mathbf{s}}) = \int f_{\mathbf{y}}(t) \log \frac{f_{\mathbf{s}}(t)}{f_{\mathbf{y}}(t)} d\mathbf{y}$$

where $f_{\mathbf{y}}$ and $f_{\mathbf{s}}$ are the probability density functions (pdfs) of $\mathbf{y}$ and $\mathbf{s}$. The pdfs are approximated by truncation of the Gram-Charlier expansion. The following learning rule is then obtained:

$$\mathbf{Q} = \mathbf{I} - \mathbf{g}(\mathbf{y}(n))\mathbf{y}^T(n), \tag{2.1}$$

$$\mathbf{W}(n+1) = \mathbf{W}(n) + \eta(n)\mathbf{Q}\mathbf{W}^T(n), \tag{2.2}$$

where $\mathbf{I}$ is the identity matrix, $\eta(n)$ is the learning rate, and $\mathbf{g}(\mathbf{y}) = (g(y_1), \cdots, g(y_n))^T$ is a nonlinear function [31],

$$g(z) = \frac{1}{2}z^5 + \frac{2}{3}z^7 + \frac{15}{2}z^9 + \frac{2}{15}z^{11} - \frac{112}{3}z^{13} + 128z^{15} - \frac{512}{3}z^{17}. \tag{2.3}$$

Since natural signals are usually super-Gaussian, we can also simply use $2\tanh(z)$ as the nonlinear function $g(z)$ when applying learning rules (2.1) and (2.2) to separate speech or music signals [39].

Furthermore, based on [2] we can derive a nonholonomic version of the learning rule that is suitable for on-line signal separation. In the nonholonomic version, the diagonal elements of $\mathbf{Q}$ are set to zero.

2.3 Subband-based ICA

Many psychoacoustic experiments have shown that humans perceive and process acoustic signals in different frequency bands independently [1, 43]. Inspired by these discoveries, we propose a new algorithm, namely, subband-based ICA, that integrates ICA with time-frequency analysis to separate mixed signals. Subband-based ICA and the early auditory models are compared in Figure 2.1. The new algorithm can accomplish the separation task successfully in the presence of strong noise, or when working in an on-line version.

The outline of the algorithm is described in the following:

1. First, each component $x_j(n)$ of the observation $\mathbf{x}(n)$, where $1 \leq j \leq m$, is filtered into subband signals.

   Though digital filter banks have been built to mimic the subbanding function of the cochlea [68], for simplicity and to provide the linearity required by ICA, the orthogonal Daubechies wavelet packet decomposition [19] is used instead of the cochlear filter bank:

   $$x_j^k(n) = < x_j^{n,N}, e_k^N >, \tag{2.4}$$

   where $x_j^{n,N} = (x_j(n), x_j(n-1), \cdots, x_j(n-N+1))$, $e_k^N = (e_k(1), e_k(2), \cdots, e_k(N))$ is a vector of coefficients determined by the k^{th} band Daubechies wavelet filter, and N is a window size.

2. The averaged powers of the decomposed signals in every band are computed and sorted by a fast sorting algorithm, for example heap sorting.

3. Then the nonholonomic learning rule (i.e., (2.1) and (2.2) with the diagonal elements of $\mathbf{Q}$ being zeros) is applied only to the bands that have the strongest power, for example, to the strongest fourth of all the signal bands, for the following reasons:

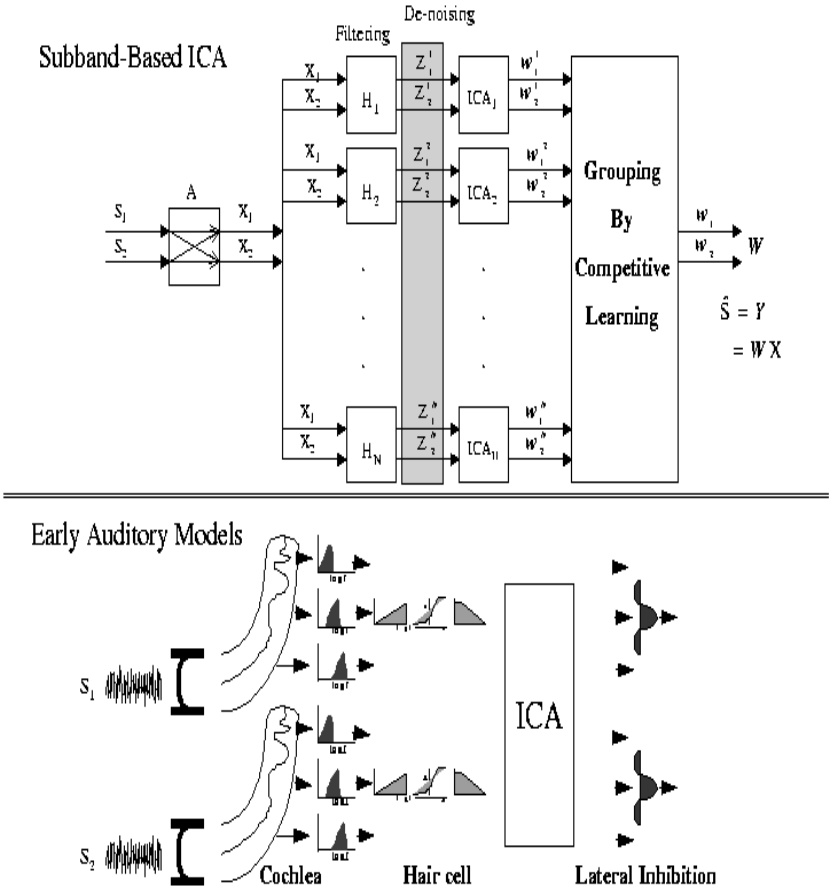

Figure 2.1: Subband-based ICA and early auditory models

- If the noise is broad-band, the signal to noise ratio (SNR) will be larger for those bands which have the strongest power.
- If the noise is limited to narrow bands, many signal bands will be noiseless, which means that good separation results can be obtained in those bands.

We denote the demixing matrix in the k^{th} selected band by

$$\mathbf{W}^k = \begin{pmatrix} \mathbf{W}_1^k \\ \vdots \\ \mathbf{W}_n^k \end{pmatrix}$$

where row $\mathbf{W}_j^k, 1 \leq j \leq n$, is used to get to the j^{th} component, $y_j^k(t)$, of the estimated source signal $\mathbf{y}(t)$ in the k^{th} band.

4. Noise is reduced using a soft thresholding algorithm [22] applied to the subband decomposed signals.

5. To recover the estimated source signal $\mathbf{y}(t)$, we have two methods:

   a. First recover the overall unmixing matrix $\mathbf{W}$ from the unmixing matrices associated with different subbands, and then recover $\mathbf{y}(t)$ from $\mathbf{W}\mathbf{x}(t)$. Competitive learning [31] is applied to cluster the rows of the unmixing matrices obtained in different subbands. The overall unmixing matrix $\mathbf{W}$ consists of n clustered rows.

   b. Recover $\mathbf{y}(t)$ directly from the $y_j^k(t), 1 \leq j \leq n$ by the wavelet packet reconstruction algorithm.

   Depending on the practical situation, we can choose (a) or (b) to get the best result.

Note that besides the virtually increased SNR in those frequency bands where we applied the ICA learning rules, the fact that subband signals, i.e., wavelet coefficients, are more peaky and heavy-tailed distributed than the original signals also greatly contributes to the success of subband based ICA when it is applied to noisy mixtures or performed on-line. Indeed, an assumption underlying the ICA learning rule is that the source signals are non-Gaussian. However, the presence of noise makes the signal mixture more like a Gaussian. Also, even with a little noise, in a short time period, the mixture signal distribution may come close to a Gaussian because of nonstationarity. Therefore classical ICA algorithms do not work well in very noisy situations when performed in an on-line version. On the other hand, wavelet coefficients of signals are much sparser than the original signals, which leads to a more peaky and heavy-tailed distribution. Actually, wavelet coefficients have been modeled by a typical super-Gaussian distribution, a Laplace distribution, in wavelet denoising and coding research [61]. Our simulations on speech and music signals also prove this point. By applying the learning rule to the super-Gaussian subband signals, subband-based ICA converges to the unmixing matrix quickly even in the case of noisy mixtures or when performed on-line.

2.4 Adaptive Basis Selection in Wavelet/DCT Packets

Subband-based ICA enhances the separation capability by decomposition of the signal into different frequency bands. But the problem of designing the filter bank remains. For example, it is desirable that we do not split the signal into two bands at the frequency where the energy

of the signal is concentrated, because otherwise we might segment one or several continuous signal streams in the time-frequency plane into two different bands, which could affect the performance of ICA in each band. So, depending on different signal properties, we can design different filter banks to improve the performance of subband-based ICA.

To address this problem, we incorporated the adaptive basis selection algorithm, proposed by Coifman et al. [15], into the subband-based ICA algorithm.

As in the procedure described in Section 2.3, we have the following steps:

1. First we choose Shannon entropy as the cost function and apply the adaptive basis selection algorithm using Wavelet or DCT packets (see the details in [15]) to the summation of the mixed signals to get the best basis.

2. Then we project each mixed signal onto the best basis.

3. The learning rule is applied only to those of the projected signals that have the strongest normalized power. Noise is reduced by thresholding if necessary.

4. Competitive learning is used to group the rows of the unmixing matrices obtained from different bases to get the overall unmixing matrix W.

The best basis selection algorithm actually accomplishes the task of adaptively selecting filter banks based on the properties of the signal, which makes subband-based ICA more robust against noise.

2.5 Experimental Results

First let us introduce a performance index E, which is defined as in [4]:

$$E = \sum_{i=1}^{n}(\sum_{j=1}^{n}\frac{|p_{ij}|}{max_k|p_{ik}|} - 1) + \sum_{j=1}^{n}(\sum_{i=1}^{n}\frac{|p_{ij}|}{max_k|p_{kj}|} - 1)$$

where $\mathbf{P} = \{p_{ij}\} = \mathbf{WA}$. The smaller the index is, the better $\mathbf{P}$ approximates a permutation matrix which has only one nonzero element in each row and each column, and the better the separation is.

In the following paragraphs, we report our experimental results both in batch and on-line modes.

In batch mode, we separated two mixtures of two speech signals, randomly selected from the TIMIT speech library, and added strong white noise. These speech signals were sampled at 8KHz. The average SNR of the mixtures was 0.51 dB. From the mixtures it was hard to understand any word of the speech. Then subband-based ICA was applied to separate the mixture signals. The performance index E of this separation was 0.08 and the SNR increased to 5.64 dB. The separated speech signals were understandable, though still noisy.

Next, still in batch mode, we tested our algorithm on two mixtures of strong white noise and the test data street.wav and beet.wav which were used at the ICA 1999 conference [38]. The power of the noise was the same as the average mixed signal power, i.e., the average SNR was 0.0 dB. Despite the low SNR, subband-based ICA based on adaptive basis selection was successful in the separation. For purposes of comparison, we also tested the Fast ICA algorithm [34] and the Extended Infomax algorithm [39] on those noisy mixtures. The codes for Fast ICA and Extended Infomax were downloaded from [33] and [41] respectively. For Extended Infomax we

Approach	Index E	Average SNR of the separated signals
Subband based ICA	0.051	4.31 dB
Fast ICA	0.124	-1.63 dB
Extended Infomax	0.118	-1.38 dB

Table 2.1: Simulation of different ICA algorithms. The average SNR of the mixed signal is 0.0 dB.

modified the learning rate trying to get the best performance for our test data. The separation results are shown in Table 2.1.

From the above table, we can see that subband-based ICA is robust against noise. The waveforms in the separation obtained using subband-based ICA are shown in Figure 2.2.

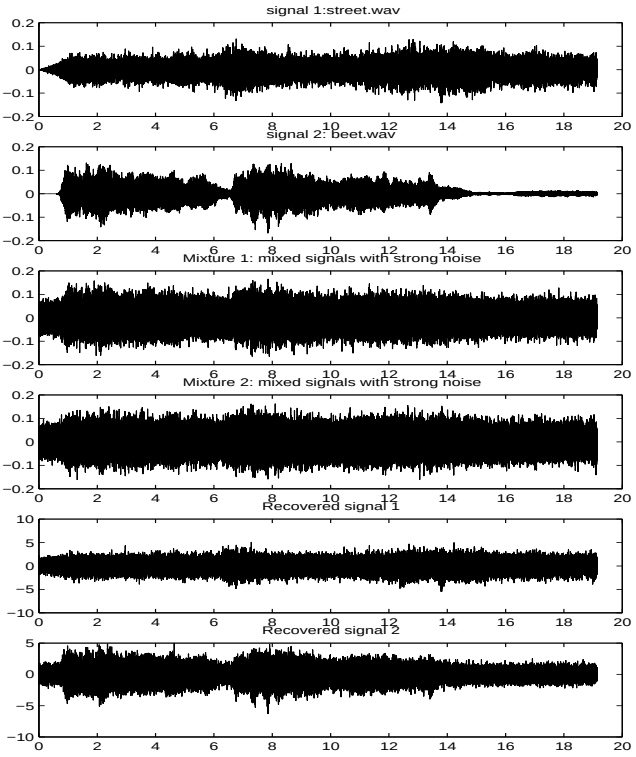

Figure 2.2: Separation in the presence of strong white noise

Our on-line separation experiments were as follows.

First, we on-line separated two mixtures of a violin melody and a segment of some pop music. These musical signals were sampled at 8K Hz. We used a modified Extended Infomax algorithm [39], nonholonomic ICA without wavelet decomposition, and subband-based ICA. We modified the Extended Infomax algorithm into an on-line version and changed its learning rate to achieve good performance on our test data. The performance indexes of these three algorithms are shown in Figure 2.3. From this figure, we can see that subband-based ICA did the separation successfully, while the other two methods failed.

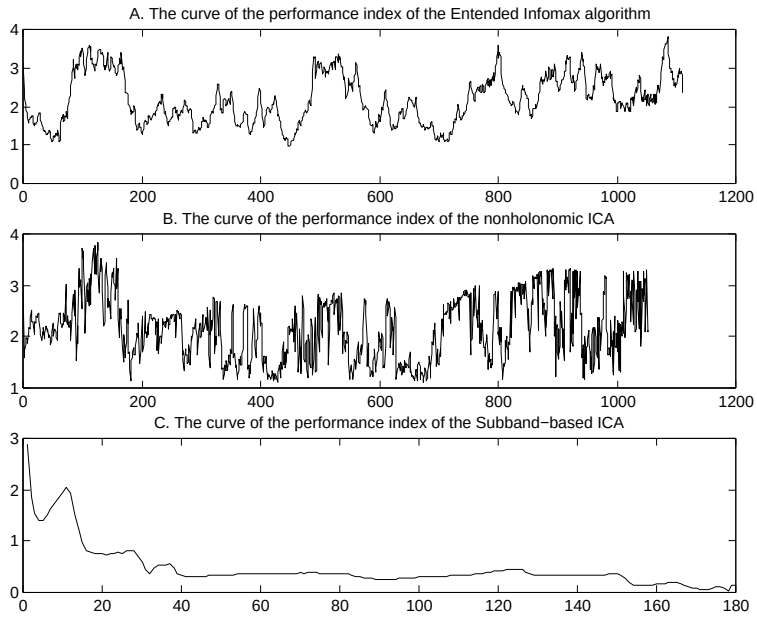

Figure 2.3: Experiment 1: the curves of the performance index E

Also, subband-based ICA was much faster than the other two methods. We used a Sun Ultra10 with 500M memory to run the Matlab scripts. The times needed to separate the mixtures are listed in Table 2.2.

Approach	Separation Time (sec.)
Modified Extended Infomax	61.72
Nonholonomic ICA	86.68
Subband-based ICA	18.05

Table 2.2: Experiment 1: The separation times needed by different ICA algorithms

Second, we tested those online algorithms on mixtures of two songs. Those signals were also sampled at 8KHz. The same three separation algorithms were tested as before. The curves of the performance index E are shown in Figure 2.4. Clearly, subband-based ICA is much better than the other two methods.

In addition, the times needed to separate the mixtures are listed in Table 2.3.

Approach	Separation Time (sec.)
Modified Extended Infomax	1582.62
Nonholonomic ICA	563.24
Subband-based ICA	101.78

Table 2.3: Experiment 2: The separation times needed by different ICA algorithms

Third, we tested the on-line algorithms on the mixtures of two other musical signals. After processing the data, the performance index E of the subband-based ICA converged to 0.0181

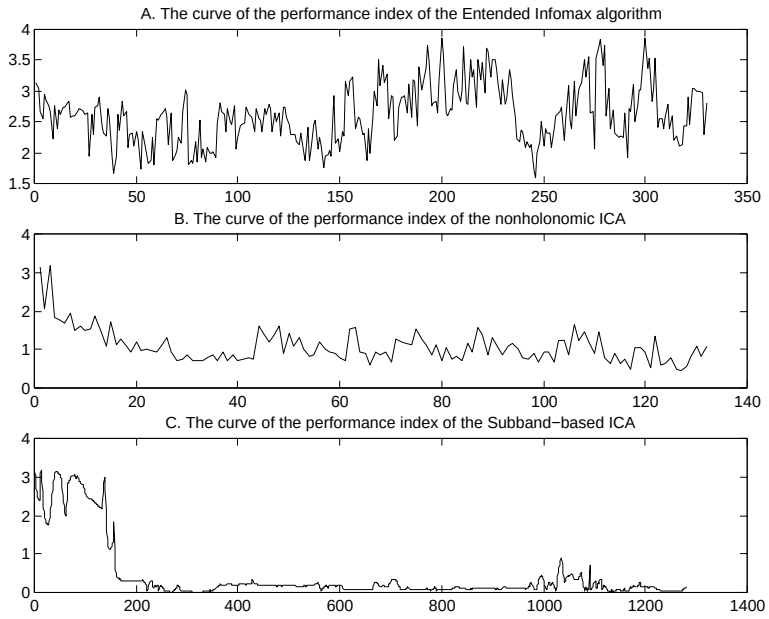

Figure 2.4: Experiment 2: the curves of the performance index E

while the performance indexes of the Extended Infomax algorithm the nonholonomic ICA were still 3.1894 and 2.1132 respectively. The waveforms of the source signals and separated signals are shown in Figure 2.5.

2.6 Conclusion

Inspired by our understanding of the subbanding strategies used in the early auditory system, we presented subband-based ICA, a powerful new algorithm for separating mixed signals. By performing separation in several frequency bands where the SNR is higher than in the original signal mixtures, subband-based ICA is robust against noise and converges to the real demixing matrix quickly. Furthermore, by incorporating a best basis selection algorithm, it can be adaptive to the properties of the signal and noise. Finally, the fact that subband signals, i.e., wavelet coefficients, are more peaky and heavy-tailed distributed than the original signals also contributes to the success of subband based ICA. The experimental results fully demonstrate its effectiveness.

Also, subband-based ICA is a computationally efficient algorithm because it reduces the computational complexity by performing separation on the down-sampled signals in several or even a single frequency band. Its speed is much higher than those of previous ICA algorithms.

Furthermore, we can generalize subband-based ICA by replacing the subband decomposition with some appropriate projection. For example, a nonlinear projection can be used under some criterion, e.g., maximum likelihood, to derive a nonlinear ICA.

Our future work will include using some signal cues, for example, the pitch of acoustic signals, and available prior knowledge, to guide separation. In this way, we may increase the convergence speed and accomplish the separation even in cases where the number of sensors is less than the number of sources. Some work has been initiated in this direction.

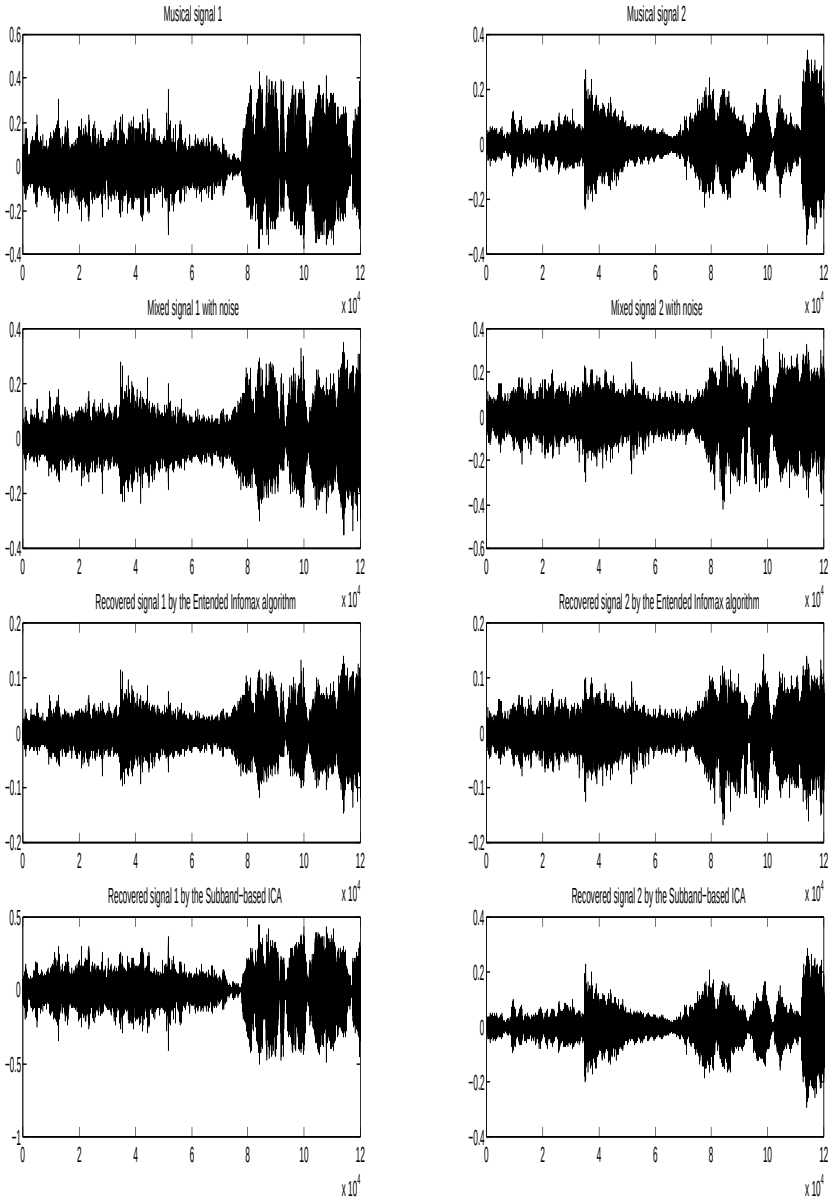

Figure 2.5: Separation of two musical signals

Chapter 3

Support Vector Machines for Pattern Recognition

3.1 Introduction

The Support Vector Machine (SVM) is a powerful new machine learning algorithm, which is rooted in statistical learning theory [65]. By constructing a decision surface hyperplane which yields the maximal margin between positive and negative examples, SVM approximately implements the *Structural Risk Minimization (SRM)* Principle. This principle is based on the fact that the error rate of a learning machine on test data is bounded by the sum of the training error rate and a second term that depends on the *Vapnik-Chervonenkis (VC) dimension*, a very important concept presented in [65]. SVM can minimize the training error rate and the second term at the same time. Many experiments have shown the good generalization performance of SVM on problems such as isolated handwritten digit recognition [58], object recognition [10], speaker identification [57], and face detection [49].

In the following sections, we first review the theories related to SVM, including the Empirical Risk Minimization Principle and the Structural Risk Minimization Principle. We then describe how the Structural Risk Minimization Principle is approximately implemented by SVM, and finally summarize and discuss properties of SVM.

3.2 Empirical Risk Minimization

3.2.1 Expected Risk and Empirical Risk

In two-class pattern recognition, the supervised learning task can be formulated as follows: Given a set of decision functions

$$f(\mathbf{x}, \lambda) : \lambda \in \Lambda, \qquad f(\mathbf{x}, \lambda) : \mathbf{R}^N \rightarrow \{-1, 1\} \tag{3.1}$$

where Λ is a set of abstract parameters, and a set of examples

$$(\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2) \ldots, (\mathbf{x_l}, y_l) \qquad \mathbf{x_i} \in \mathbf{R}^N, y_i \in \{-1, 1\}$$

drawn from an unknown distribution $P(\mathbf{x}, y)$, find a function $f(\cdot, \lambda^*)$ that provides the smallest possible value for the expected risk:

$$R(\lambda) = \int \frac{1}{2} |f(\mathbf{x}, \lambda) - y| P(\mathbf{x}, y) d\mathbf{x} dy$$

The functions $f(\cdot, \lambda)$ are called hypotheses, and the set $\{f(\cdot, \lambda) : \lambda \in \Lambda\}$ is called the hypothesis space and is denoted by H. Thus the expected risk is a measure of how good a hypothesis is

at predicting the correct value y at a point $\mathbf{x}$. The function $f(\cdot, \lambda)$ is called a *trained machine*, given a particular choice of λ through training. For example, the hypotheses could be Radial Basis Functions or Multi-layer Perceptrons with a fixed structure. In this case, the parameter set Λ is the set of weights and biases of the networks.

Because the probability distribution $P(\mathbf{x})$ is unknown, it is impossible to compute the expected risk $R(\lambda)$ directly. So instead of trying to get the exact value of $R(\lambda)$, a statistical approximation of $R(\lambda)$, called the empirical risk, is computed on the training set as follows:

$$R_{emp}(\lambda) = \frac{1}{2l} \sum_{i=1}^{l} |f(\mathbf{x_i}, \lambda) - y_i|$$

3.2.2 Uniform Convergence and VC Dimension

According to the law of large numbers, the empirical risk R_{emp} converges in probability to the expected risk R. Hence a straightforward idea is minimizing the empirical risk rather than the expected risk. This idea is called the *Empirical Risk Minimization (ERM)* Principle. An assumption in the ERM Principle is that if R_{emp} is converging to R, the minimum of R_{emp} will converge to the minimum of R too. If this assumption actually does not hold, the ERM Principle does not lead to a correct inference. Fortunately, as shown by Vapnik and Chervonenkis [64], this assumption holds if and only if convergence in probability of R_{emp} to R is replaced by *uniform* convergence in probability. Here, uniform convergence is defined as follows:

$$\text{for any } \lambda \in \Lambda \text{ and } \epsilon > 0,\ P(\sup_{\lambda} |R(\lambda) - R_{emp}(\lambda)| > \epsilon) \to 0 \qquad \text{as } N \to \infty$$

Vapnik and Chervonenkis also showed that the finiteness of the VC dimension h of hypothesis space H is the necessary and sufficient condition for uniform convergence of the ERM. The VC dimension of the hypothesis space H is defined as follows:

> Consider functions that correspond to the two-class pattern recognition case as defined in (3.1). If a given set of l points can be labeled in all 2^l possible ways, and for each labeling, a member of the set $\{f(\cdot, \lambda)\}$ can be found which correctly assigns those labels, we say that that set of points is *shattered* by that set of functions. The VC dimension for the set of functions $\{f(\cdot, \lambda)\}$ is defined as the maximum number of training points that can be shattered by $\{f(\cdot, \lambda)\}$. In [11], it is proved that the VC dimension of the set of oriented hyperplanes in $\mathbf{R}^N$ is $N + 1$.

Thus the VC dimension is a measure of the complexity of H, and it is often, but not necessarily, related to the number of free parameters of $f(\cdot, \lambda)$). For example, the VC dimension of a set of Radial Basis Functions or Multi-layer Perceptrons is controlled by the number of hidden units.

3.2.3 Risk Bound

Using the concept of the VC dimension, Vapnik [65] derives a bound on the deviation of the empirical risk from the expected risk. That is, with probability $1 - \eta$ where $0 \leq \eta \leq 1$, the following inequality holds:

$$R(\lambda) \leq R_{emp}(\lambda) + \sqrt{\frac{h(\log(2l/h) + 1) - \log(\eta/4)}{l}} \qquad \forall\, \lambda \in \Lambda \tag{3.2}$$

where h is the VC dimension of $f(\cdot, \lambda)$), the right hand side of (3.2) is called the "risk bound", and the second term on the right side is called the "VC confidence". This bound is independent

of $P(\mathbf{x}, y)$. Clearly, in order to achieve small expected risk, which means good generalization performance, both the empirical risk and the VC confidence have to be small. Because the VC confidence is an increasing function of the VC dimension h and R_{emp} is usually a decreasing function of h, there is a tradeoff between these two terms when choosing the value of h. How to choose an appropriate value for h is a difficult but important problem.

3.3 Structural Risk Minimization

The bound (3.2) suggests a new induction principle, *Structural Risk Minimization (SRM).*

The SRM Principle of Vapnik [64] aims to solve the problem of choosing an appropriate VC dimension. Note that while the VC confidence depends on the VC dimension h of the given class of functions, the empirical risk R_{emp} depends on the particular function chosen in training. To minimize h and R_{emp} at the same time, Vapnik constructs a nested structure of hypothesis spaces

$$H_1 \subset H_2 \subset \cdots \subset H_n \subset \cdots$$

with the property that $h(n) \leq h(n+1)$ where $h(n)$ is the VC dimension of the set H_n and can be computed, or has an upper bound. For each set H_n, the goal of the training is simply to minimize R_{emp}. Then the trained machine whose sum of VC dimension and R_{emp} is minimal among all trained machines is chosen as the final learning machine. SVM approximately implements the SRM Principle so that the VC dimension and R_{emp} are minimized at the same time.

3.4 Construction of Support Vector Machines

This section describes how to construct a Support Vector Machine (SVM) [65] step by step, from the simplest case of linearly separable patterns to linearly non-separable patterns, and finally to non-separable patterns.

3.4.1 Optimal Hyperplane for the Linearly Separable Case

First, in the linearly separable case, one wishes to find the best hyperplane that separates the data. Here, "linear separable" means that one can find a pair $(\mathbf{w}, b)$ such that

$$\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b \geq 1 \qquad \forall\, \mathbf{x}_i \in \text{Class 1} \tag{3.3}$$

$$\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b \leq -1 \qquad \forall\, \mathbf{x}_i \in \text{Class 2} \tag{3.4}$$

In this case, the hypothesis space is the set of functions

$$f(\mathbf{x}; \mathbf{w}, b) = \text{sign}(\mathbf{w}^{\mathbf{T}}\mathbf{x} + b) \tag{3.5}$$

To make a decision surface correspond to a unique parameter pair $(\mathbf{w}, b)$, the following constraint is imposed:

$$\min_{i=1,\ldots,l} |\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b| = 1 \tag{3.6}$$

where $\mathbf{x}_1, \ldots, \mathbf{x}_l$ are points in the data set. The hyperplanes that satisfy (3.6) are called *canonical hyperplanes.* Notice that (3.6) is just a normalization. As mentioned in Section 3.2.2, the VC dimension of the canonical hyperplanes in $\mathbf{R}^N$ is $N + 1$, which is the total number of free parameters in (3.5). To implement the SRM Principle, a structure on the set of canonical hyperplanes is produced by adding another constraint as follows:

Let D denote the diameter of the smallest N-dimensional sphere containing all the points $\mathbf{x}_1, \ldots, \mathbf{x}_l$. Then the set

$$f(\mathbf{x}; \mathbf{w}, b) = \text{sign}(\mathbf{w}^\mathbf{T}\mathbf{x} + b) \mid \|\mathbf{w}\| \leq A \tag{3.7}$$

has VC dimension h that satisfies the following bound [65]:

$$h \leq \min \lceil D^2 A^2 \rceil, N + 1 \tag{3.8}$$

Moreover, it can be shown that the distance from a point $\mathbf{x}$ to the hyperplane defined by the pair $(\mathbf{w}, b)$ is

$$d(\mathbf{x}; \mathbf{w}, b) = \frac{|\mathbf{w}^\mathbf{T}\mathbf{x} + b|}{\mathbf{w}} \tag{3.9}$$

Substituting (3.6) into (3.9), it follows that the distance between the canonical hyperplane and the closest data point is $\frac{1}{\|\mathbf{w}\|}$. Thus if $\|\mathbf{w}\| \leq A$, the distance between the canonical hyperplane and the closet data point has to be larger than $\frac{1}{A}$. This means that the constrained set of canonical hyperplanes of (3.7) is the set of hyperplanes whose distance from the data points is at least $\frac{1}{A}$. Clearly, after the normalization, the distance between the two classes is $\frac{2}{\|\mathbf{w}\|}$. This distance is called the *margin of separation*.

According to the bound (3.8), minimizing $\|\mathbf{w}\|$ will make the VC dimension small. So among the canonical hyperplanes that correctly classify the data, the one with the smallest $\|\mathbf{w}\|$ minimizes the risk bound (3.2). Formally, finding the optimal decision plane is equivalent to the following quadratic programming (QP) problem:

$$\begin{aligned} \text{Minimize} \quad & \Phi(\mathbf{w}) = \tfrac{1}{2}\|\mathbf{w}\|^2 \\ \mathbf{w}, b & \\ \text{subject to} \quad & y_i(\mathbf{w}^\mathbf{T}\mathbf{x}_i + b) \geq 1 \quad i = 1 \ldots l \end{aligned} \tag{3.10}$$

This constrained optimization problem is called the *primal problem*. Here the cost function $\Phi(\mathbf{w})$ is a convex function of $\mathbf{w}$ and the constraints are linear in $\mathbf{w}$.

This problem can be solved by the technique of Lagrange Multipliers. The Lagrangian function is constructed as follows:

$$L(\mathbf{w}, b, \mathbf{\Lambda}) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{l} \lambda_i [y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1] \tag{3.11}$$

where $\mathbf{\Lambda} = (\lambda_1, \ldots, \lambda_l)$ is the vector of non-negative Lagrange multipliers (notice that here the definition of $\mathbf{\Lambda}$ is different from that in (3.1)). The solution to this optimization problem is determined by the saddle point of $L(\mathbf{w}, b, \mathbf{\Lambda})$, which has to be minimized with respect to $\mathbf{w}$ and b, and maximized with respect to $\mathbf{\Lambda} \geq 0$. By differentiating $L(\mathbf{w}, b, \mathbf{\Lambda})$ with respect to $\mathbf{w}$ and b, it follows that

$$\mathbf{w} = \sum_{i=1}^{l} \lambda_i y_i \mathbf{x}_i \tag{3.12}$$

$$\sum_{i=1}^{l} \lambda_i y_i = 0 \tag{3.13}$$

The solution vector $\mathbf{w}$ is defined in terms of a linear combination of training vectors. From the training procedure the optimal $\mathbf{w}^*$ can be explicitly and uniquely determined by virtue of the

convexity of the Lagrangian. To determine the optimal b^*, however, one needs to resort to the *Karush-Kuhn-Tucker (KKT)* "complementary" condition [27]:

$$\lambda_i^* [y_i(\mathbf{w}^{*^T}\mathbf{x}_i + b^*) - 1] = 0 \qquad \text{for } i = 1, 2, \ldots, l \tag{3.14}$$

Only those Lagrange multipliers exactly satisfying (3.14) can assume nonzero values. The data points $(\mathbf{x}_i, y_i)$ for which the corresponding $\lambda_i^* > 0$ are called *support vectors*. From a geometric perspective, the support vectors are those data points that lie closest to the decision surface. From (3.14) it follows that the optimal b^* can be computed as

$$b^* = y_i - \mathbf{w}^{*^T}\mathbf{x}_i$$

for any support vector. In practice, it is numerically safer to take the mean value of all such b^*s.

From (3.12) and (3.13), the original primal problem can be reformulated into its dual problem:

$$\begin{aligned} \text{Maximize} \qquad & Q(\mathbf{\Lambda}) = \textstyle\sum_{i=1}^{l} \lambda_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \lambda_i \lambda_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \\ \text{subject to} \qquad & \textstyle\sum_{i=1}^{l} \lambda_i y_i = 0 \\ & \lambda_i \geq 0 \qquad \text{for } i = 1, 2, \ldots, l \end{aligned} \tag{3.15}$$

Also, we can reformulate the decision function (3.5) as

$$f(\mathbf{x}) = \text{sign}\left(\sum_{i=1}^{l} y_i \lambda_i \mathbf{x}^T \mathbf{x}_i + b \right) \tag{3.16}$$

where $(\mathbf{x}_i, y_i)$ are support vectors.

3.4.2 Soft Margin Hyperplane for the Linearly Non-Separable Case

In the linearly non-separable case, there exists at least one data point $(\mathbf{x}_i, y_i)$ that violates the constraint:

$$y_i(\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b) \geq 1, \qquad i = 1, 2, \ldots, l$$

Accordingly, the margin of separation is said to be *soft*. To deal with the non-separable case, one needs a new set of nonnegative scalar variables,$\{\xi_i\}_{i=1}^{l}$, defined as follows:

$$y_i(\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b) \geq 1 - \xi_i, \qquad i = 1, 2, \ldots, l \tag{3.17}$$

$\{\xi_i\}_{i=1}^{l}$ measure the oriented distance of a data point to the decision hyperplane. When $\xi_i \geq 1$, the data point falls on the wrong side of the decision hyperplane. In this case, the support vectors are the data points that satisfy (3.17) with equality even if $\xi_i > 0$.

The following function measures the total number of misclassifications:

$$\Phi(\xi) = \sum_{i=1}^{l} I(\xi_i - 1)$$

where the indicator function $I(\xi)$ is defined by

$$I(\xi) = \begin{cases} 0 & \text{if } \xi \leq 0 \\ 1 & \text{if } \xi > 0 \end{cases}$$

Unfortunately, using the indicator function in $\Phi(\xi)$ results in a nonconvex optimization problem that is NP-complete. To make the optimization problem tractable, $\Phi(\xi)$ is approximated by

$$\Phi(\xi) = \sum_{i=1}^{l} \xi_i$$

Finally in order to maximize the margin and minimize the number of misclassifications simultaneously, SVM solves the following primal problem:

$$\begin{aligned}
\text{Minimize} \quad & \Phi(\mathbf{w}, \xi) = \tfrac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{l} \xi_i && (3.18)\\
\mathbf{w}, b, \xi & \\
\text{subject to} \quad & y_i(\mathbf{w}^{\mathbf{T}}\mathbf{x}_i + b) \geq 1 - \xi_i \quad i = 1 \ldots l && (3.19)\\
& \xi_i \geq 0 \quad i = 1 \ldots l && (3.20)
\end{aligned}$$

Minimizing the first term in (3.18) leads to minimizing the VC dimension of the learning machine, and minimizing the second term controls the empirical risk. Therefore, this approach constitutes an approximate implementation of the SRM principle. Here the parameter C controls the tradeoff between the complexity and the empirical risk of of the trained machine.

As in the previous section, the dual problem can be formulated as

$$\begin{aligned}
\text{Maximize} \quad & Q(\mathbf{\Lambda}) = \sum_{i=1}^{l} \lambda_i - \tfrac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} \lambda_i\lambda_j y_i y_j \mathbf{x_i}^T\mathbf{x}_j \\
\text{subject to} \quad & \sum_{i=1}^{l} \lambda_i y_i = 0 \\
& 0 \leq \lambda_i \leq C \qquad \text{for } i = 1, 2, \ldots, l && (3.21)
\end{aligned}$$

The dual problem for the case of non-separable patterns differs from that for the simple case of linearly separable patterns in a minor but important way: the constraint $\lambda_i \geq 0$ is replaced by the more stringent constraint $0 \leq \lambda_i \leq C$. Except for this, the optimization for the non-separable case and the computation of the optimal $\mathbf{w}^*$ are done in the same way as in the linearly separable case.

In addition, the optimal bias value is computed in a way similar to that described before. Specifically, from the KKT conditions it follows that

$$\lambda_i^* \left[y_i(\mathbf{w}^{*^T}\mathbf{x}_i + b^*) - 1 + \xi_i\right] = 0 \qquad \text{for } i = 1, 2, \ldots, l \tag{3.22}$$

To make all the variables $\{\xi_i\}_{i=1}^{l}$ nonnegative, it can be derived from the Lagrange technique that

$$\mu_i \xi_i = 0, \qquad i = 1, 2, \ldots, l \tag{3.23}$$

where the μ_i are the Lagrange multipliers. Setting the derivative of the Lagrangian function for the primal problem with respect to the variable ξ_i to zero leads to

$$\lambda_i + \mu_i = C \tag{3.24}$$

From (3.23) and (3.24), it follows that

$$\mu_i = 0 \quad \text{if } \lambda_i < C \tag{3.25}$$

Hence the optimal bias b^* can be computed by taking any data point $(\mathbf{x}_i, y_i)$ in the training set for which we have $0 \leq \lambda_i \leq C$ and therefore $\mu_i = 0$, and using that data point in (3.22). As mentioned before, it is numerically safer to take the mean value of b^* resulting from all such training data.

3.4.3 Nonlinear Decision Surfaces

This section extends the linear optimal hyperplane to more complicated decision surfaces for the real-world pattern recognition problem. The extension involves two operations:

- First, nonlinearly map an input variable $\mathbf{x}$ into a high-dimensional *feature space*.
- Then, construct an optimal hyperplane in the high-dimensional feature space.

The first operation is justified by *Cover's theorem* on the separability of patterns, which may be stated as follows [17]:

> "A complex pattern-classification problem cast in a high-dimensional space nonlinearly is more likely to be linearly separable than in a low-dimensional space."

In the second operation, an optimal hyperplane is built in the same way as described in the previous sections, except that the support vectors are not drawn from the input space, but from the high-dimensional feature space.

Let $\{\varphi_j\}_j^M$ denote a set of nonlinear transformations from the input space to the feature space, where M is the dimension of the feature space. The nonlinear mapping is defined as

$$\mathbf{x} \rightarrow \varphi(\mathbf{x}) = (\varphi_1(\mathbf{x}), \varphi_2(\mathbf{x}), \ldots, \varphi_M(\mathbf{x})) \tag{3.26}$$

Then a SVM is constructed as follows:

$$f(\mathbf{x}) = \text{sign}(\mathbf{w}^{*^T} \varphi(\mathbf{x}) + b^*) = \text{sign}\left(\sum_{i=1}^{l} y_i \lambda_i^* \varphi^{\mathbf{T}}(\mathbf{x}) \varphi(\mathbf{x_i}) + b^* \right) \tag{3.27}$$

Let K denote the *inner-product kernel*, which is defined as

$$K(\mathbf{x}, \mathbf{z}) = \varphi^{\mathbf{T}}(\mathbf{x}) \varphi(\mathbf{z}) = \sum_{j=0}^{M} \varphi_j(\mathbf{x}) \varphi_j(\mathbf{z}) \tag{3.28}$$

Substituting (3.28) into (3.27) yields

$$f(\mathbf{x}) = \text{sign}\left(\sum_{i=1}^{l} y_i \lambda_i^* K(\mathbf{x}, \mathbf{x}_i) + b^* \right) \tag{3.29}$$

and the QP problem (3.21) becomes

$$\begin{array}{ll} \text{Maximize} & Q(\mathbf{\Lambda}) = \sum_{i=1}^{l} \lambda_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \lambda_i \lambda_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \\ \text{subject to} & \sum_{i=1}^{l} \lambda_i y_i = 0 \\ & 0 \leq \lambda_i \leq C \qquad \text{for } i = 1, 2, \ldots, l \end{array} \tag{3.30}$$

The use of the kernel trick greatly reduces the high computational burden of the nonlinear mapping into the high-dimensional space in SVM.

Note that the expansion (3.28) is a special case of *Mercer's theorem* [44]. According to Mercer's theorem, the functions $\varphi_j(\mathbf{x})$ are eigenfunctions of the expansion. They are positive definite. In theory, the dimensionality of the feature space (i.e., the number of eigenfunctions) can be infinitely large. Mercer's theorem provides a way to check whether a candidate function is really an inner-product kernel in some space. Some commonly used inner-product kernels are listed in Table 3.1.

Inner-Product Kernel	Type of Classifier
$K(\mathbf{x}, \mathbf{z}) = \exp(-\|\mathbf{x} - \mathbf{z}\|^2)$	Radial basis function network
$K(\mathbf{x}, \mathbf{z}) = (1 + \mathbf{x}^T\mathbf{z})^d$	Polynomial Learning Machine
$K(\mathbf{x}, \mathbf{z}) = \tanh(\mathbf{x}^T\mathbf{z} - \theta)$ (only for some values of θ which satisfy Mercer's theorem)	Multi-layer Perceptron

Table 3.1: Summary of commonly used inner-product kernels

3.5 Summary and Discussion

In this section we first summarize some important properties of SVM:

- As an approximate implementation of the SRM Principle, SVM provides a method of minimizing the empirical risk and the VC dimension at the same time, so that the risk bound of the trained machine can be minimized, i.e., the trained machine has good generalization performance.
- By reformulating the primal optimization problem into its dual problem and using a suitable inner-product kernel, SVM controls the model complexity independently of the dimensionality of the feature space. Actually, an infinite feature space is allowed in SVM.
- Moreover, the convex cost function in the QP problem guarantees that SVM will find a globally optimal solution, while many other learning algorithms suffer from falling into local extrema.
- By solving the QP problem during the training phase, SVM automatically tunes all the parameters in a learning scheme.

Though originally derived from the SRM Principle to address the problem of the tradeoff between model complexity and generalization ability, SVM is closely related to other known techniques and research problems:

- The support vectors are usually *sparse*. They only constitute a fraction of the total number of examples in the training set. Using the reproducing property of the Reproducing Kernel Hilbert Space (RKHS), Girosi [29] shows an equivalence between SVMs in the noiseless case and a Sparse Approximation (SA) scheme that resembles the Basis Pursuit De-Noising algorithm [14].
- Also in [29], Girosi gives a derivation of the SVM algorithm in the framework of regularization theory. In [24], Evgeniou et al. give a a unified framework for regularization networks and SVM. The reformulation of SVM in regularization theory reveals the connection between SVM and other known techniques. However, it hides the relation between SVM and the SRM Principle.
- SVM provides high generalization performance without incorporating any prior knowledge of the problem. An important research topic is how to incorporate problem-domain knowledge into SVM to further improve its performance. Some proposed approaches include adding an additional term that represents prior knowledge in the cost function, using prior knowledge to design the kernel function [59], and adding virtual examples into the training set [58]. More efficient and natural ways of adding prior knowledge into SVM

have yet to be developed. For example, integrating Bayesian learning theory into SVM might be a good way of exploiting prior information.

- The kernel trick in SVM can also be used in other algorithms that are based on the inner product of the data. For example, Principal Component Analysis can be done in high-dimensional feature space by using a suitable nonlinear kernel function [60]. Fisher discriminant analysis also uses a similar idea [45].

Chapter 4

A Hybrid ICA/SVM Learning Scheme and its Application to Face Detection

4.1 Introduction

In this chapter, we propose a new hybrid unsupervised/supervised learning scheme that integrates Independent Component Analysis with the Support Vector Machine (SVM), and we apply this new learning scheme to the face detection problem. As a powerful unsupervised learning algorithm, ICA can not only "blindly" separate mixed signals as shown in Chapter 2, but also effectively extract low-level features in signals. In [9], Bell and Sejnowski point out that the independent components of natural scenes are edge filters. And in [48], Olshausen and Field show an equivalence between sparse coding and an ICA algorithm in the case of no noise and a square system (i.e., the dimensionalities of output and input are same). Using their feature extraction capability, ICA algorithms have been successfully used in face recognition and facial expression analysis, and have achieved better results than Principal Component Analysis (PCA) [8, 21, 42]. On the other hand, SVM is a promising supervised learning algorithm. As discussed in Chapter 3, by minimizing the empirical risk on the training data set and the model complexity, measured by the VC dimension, at the same time, SVM gives good generalization performance on pattern recognition problems. Combining these two learning algorithms yields a powerful hybrid learning scheme. In this chapter we apply this new learning scheme to the face detection problem. Specifically, we use ICA in two different ways to extract low-level features from a window sliding over an image, and then apply SVM at a high level to classify the extracted ICA features as face or not. In addition, to reduce the time of the detection procedure, a skin-color filter is implemented to find the candidate face regions in an image, so that the sliding window moves over reduced image regions. Experimental results demonstrate the effectiveness of the new hybrid learning scheme on the face detection problem.

The rest of this chapter is organized as follows. Section 4.2 gives a short review of face detection methods. Section 4.3 presents the method of finding candidate face regions in images using a skin-color filter. Section 4.4 presents the hybrid ICA/SVM learning scheme. Section 4.5 describes the face detection system based on this scheme. Section 4.6 reports our experimental results. Finally, Section 4.7 contains conclusions and discussion.

4.2 Literature Review on Face Detection

Face detection has important applications in various areas, such as intelligent human-computer interaction, video surveillance, video indexing, and object-based video coding. These applications have contributed to an increasing research interest in face detection in recent years. In

this section, we give a short review of the technical literature on face detection.

In [62] Sung and Poggio propose an example-based learning approach to detecting frontal human faces. They use six Gaussian clusters to model the distributions of face patterns and six other Gaussian clusters for non-face patterns, and use two distance metrics to train a Multilayer Perceptron as the classifier. Rowley et al., in [54] and [55], use a retinally connected neural network to detect faces in an image. Multiple networks are used to improve system performance. In [49] Osuna et al. apply a support vector machine to face detection and obtain slightly better results than Sung and Poggio on two test sets. In [50] Qian and Huang report a detection scheme that combines template matching and a feature-based detection algorithm using hierarchical Markov random fields (MRF) and maximum *a posteriori* probability (MAP) estimation. In [56] Samaria uses Hidden Markov Models (HMM) for face detection. In [16] Colmenarez and Huang use Kullback divergence to maximize the discrimination between positive and negative examples of faces. A family of discrete Markov processes is used to model faces and background patterns. Detection is based on the likelihood ratio computed during the training phase. In [46] Moghaddam and Pentland propose a detection method that is based on density estimation in a high-dimensional space using an eigenspace decomposition. In [51] Rajagopalan et al. apply higher-order statistics and HMMs to detect faces. In [53], Roth et al. present a face detection method that uses a Sparse Network of Windows (SNoW) learning architecture, which has been successfully used in the natural language domain.

In addition to these statistical methods, in [69] Yuille uses deformable templates to model facial features. In this approach, facial features are described by parameterized templates. The best fit of the elastic model is obtained by minimizing an energy function. Texture information has also been used for detecting faces [18].

To speed up the detection procedure, color and motion information can be exploited in color images and video sequences [66]. A single Gaussian or a mixture of Gaussians can be used to model the skin color distribution. Expectation-Maximization (EM), an iterative maximum-likelihood estimation algorithm, provides an effective way of learning a Gaussian mixture model [52]. More recently, several modular systems combining shape analysis, color segmentation and motion information have been used for locating and tracking faces in a video sequence [30].

In [67] Yang gives an extensive survey of face detection methods. In [13], Chellappa et al. give a comprehensive survey of the literature on human and machine recognition of faces, which is closely related to face detection.

After this short review of face detection, we are ready to present our face detection system based on the hybrid ICA/SVM learning scheme. In a preliminary section, we first introduce the preprocessing procedure used in our detection system, which includes two main components, a skin color filter and histogram equalization.

4.3 Skin Color Filter and Other Preprocessing

Though skin color can be nicely modeled as a mixture of Gaussians as mentioned before, our system uses a simpler method to find possible skin regions in an image because we only use it to reduce the search area in the image instead of finding exact face locations. This method is a modified version of the skin filter proposed in [26] and, in essence, is a thresholding approach in hue and saturation space. It includes the following steps:

- First, the input color image in RGB format is transformed to log-opponent (IRgBy) values, and from these values the amplitude, hue, and saturation are computed. The conversion

from RGB to log-opponent is computed as follows:

$$I = \frac{L(R) + L(B) + L(G)}{3} \tag{4.1}$$

$$R_g = L(R) - L(G) \tag{4.2}$$

$$B_y = L(B) - \frac{L(G) + L(R)}{2} \tag{4.3}$$

where $L(x) = 105 \log 10(x + 1)$.

- Second, the log opponents are transformed into hue-saturation space as follows:

$$H = \arctan^2(R_g / B_y) \tag{4.4}$$

$$S = \sqrt{R_g^2 + B_y^2} \tag{4.5}$$

where H and S represent the hue and saturation images respectively, and the unit for H is degrees.

Figures 4.1, 4.2, and 4.3 show a color image, its hue image, and its saturation image respectively. Note that there is a strong blocking effect in the hue image (Figure 4.2). The reason is that, in image coding, many fewer bits are assigned to the color information than to the gray intensity information. This suggests that using only color information to locate faces in images is not robust to image coding error.

Figure 4.1: A test image for face detection

- Next, by a simple thresholding method, we produce a binary mask $M_{x,y}$ that locates face candidate regions in an image. The thresholding method is defined as follows:

$$M_{x,y} = \begin{cases} 1 & \text{if } 120 < H_{x,y} < 175 \\ 0 & \text{if } 15 < S_{x,y} < 75 \end{cases}$$

where $M_{x,y}, H_{x,y}, S_{x,y}$ are the values of the binary face candidate mask, hue image, and saturation image at pixel (x, y) respectively.

Figure 4.4 shows the binary face candidate mask for the image in Figure 4.1.

It seems that the skin color filter works perfectly, as shown in Figure 4.4. However, sometimes the skin color filter does not work well. It tends to falsely detect highly saturated red and yellow areas as face candidate areas. The reason for the problem may

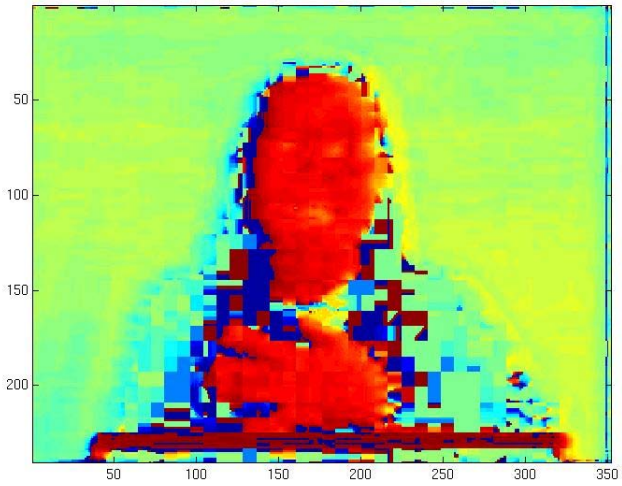

Figure 4.2: Test image in hue space

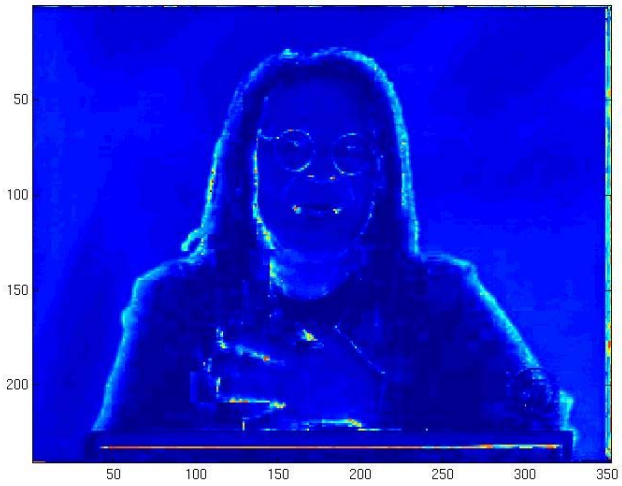

Figure 4.3: Test image in saturation space

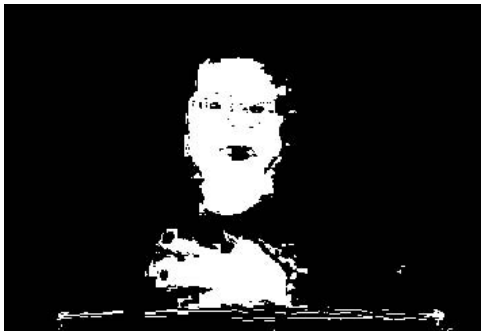

Figure 4.4: Possible face areas for test image

be an improper threshold or the low-rate image coding. Clearly, this simple approach is inadequate for finding accurate face locations in an image. However, it is acceptable for our system because we only use it as part of the preprocessing to reduce the image area scanned by a sliding window.

- After the binary face candidate mask is produced, simple morphological operations are performed on the mask, which include binary closure (i.e., dilation followed by erosion) and removal of small blobs, because small blobs usually arise from non-face regions.

  The binary mask after the morphological operations is shown in Figure 4.5.

Figure 4.5: Final face candidate mask for the test image

Using the mask, we can find candidate face areas in an image, and the sliding window can move only over the candidate areas instead of the whole image. In order to compensate for different illuminations, camera responses, etc., histogram equalization is performed over the image blocks defined by the sliding window.

4.4 Hybrid ICA/SVM Learning Scheme

In this section we present a new hybrid learning scheme that integrates ICA and SVM. By exploiting higher-order statistics, ICA can find an independent basis for the data, and obtain a better clustering and representation of the data than PCA. When applied to natural images,

ICA filters are edge filters. When training on a large number of natural image blocks, we can even get a wavelet-like ICA basis. But in contrast to wavelet analysis, the ICA basis is adaptive to the training data.

In the hybrid learning scheme, after ICA feature extraction, SVM is applied to classify the features. One common characteristic of ICA and SVM is sparsity. The ICA output is sparse, and the support vectors in SVM are also sparse. In the following section, we describe this hierarchical sparse learning architecture.

4.4.1 ICA Feature Extraction

Redundancy Reduction, ICA, and Sparse Coding

As shown in [5, 6, 7, 20], an important characteristic of sensory processing in the brain is redundancy reduction. One method of achieving redundancy reduction is based on the minimization of mutual information of the system outputs. According to the theory developed in Chapter 2, we know that ICA is such an algorithm. Actually, experimental results have shown that trained ICA bases are very similar to the receptive fields of simple cells in mammalian visual cortex [9, 47, 48]. [63] reports a detailed comparison between ICA features and the properties of simple cells in the macaque primary visual cortex, and finds good matches to most of the parameters, especially if video sequences are used instead of still images.

Another method of reducing redundancy is sparse coding [7, 25, 47], which adds to the cost function a term that represents the sparseness of the output. If the data has a super-Gaussian distribution, sparse coding results in approximate redundancy reduction. These two approaches are equivalent to each other in some cases, as shown in [48].

Connection between Projection Pursuit and ICA Feature Extraction

In addition to its close relation to sparse coding, ICA feature extraction is related to *projection pursuit* [32]. Projection pursuit tries to find "interesting" projections for multidimensional data. In [32] Huber argues that the most interesting directions are those that show the least Gaussian distributions. At the same time, ICA can be interpreted as a search for a projection such that the unmixed signals have maximal non-Gaussianity [35]. The use of the same criterion in projection pursuit and ICA reveals the connection between these techniques.

Two Methods of ICA Feature Extraction

Given the ICA generative model under the noise-free assumption

$$\mathbf{x} = A\mathbf{s}$$

and the ICA reconstruction model

$$\mathbf{y} = W\mathbf{x},$$

there are two different methods of extracting features using ICA. These two methods were proposed by Bartlett et al. [8] for face recognition and were shown to provide equally good recognition performance. The first method is to find the statistically independent basis images by ICA and then represent the image by the coefficients of the projection on those basis images [8]. The second method is to first reduce the dimensionality of the image, and then represent the image by the independent coefficients that are obtained by applying ICA [8, 42]. These two methods are explained in detail in what follows:

Method 1: Independent Image Basis

In Method 1, each row component in the mixture $\mathbf{x}$ represents a training image and each row component in $\mathbf{y}$ represents an independent image basis. Note that here an image is represented by a vector that is the concatenation of all the columns in the image. An ICA representation of an image is the vector of coefficients of the projection on the independent bases in $\mathbf{y}$.

Because the dimensionality of $\mathbf{y}$ is the same as that of $\mathbf{x}$, it becomes necessary to control the number of independent bases when the size of the training set is very large. Since we assume the images in $\mathbf{x}$ are linear combinations of unknown independent sources in the ICA generative model, we do not lose information by replacing the original images with their m linear combinations.

If we view all the pixels in an image as an observation of a random vector, then the images are linear combinations of the eigenvectors of their covariance matrix. We choose those eigenvectors that correspond to the m maximal eigenvalues as the ICA training images, because they contain most of the energy of the original image set. Denote these eigenvectors by p_i where $i = 1, \ldots, m$. Then p_i is a $N \times 1$ column vector, where N is the pixel number in the image. Denote the matrix that contains the m column vectors, p_i, by P_m. By performing ICA on P_m^T, we can obtain a matrix of m independent source images. Note that here p_m^T is a row in $\mathbf{x}$. Formally, we have the following steps:

First from the matrix $\mathbf{x}$, we find the eigenvector matrix P_m. Then we take P_m^T as the mixture $\mathbf{x}$ and apply the ICA algorithm as follows:

$$\begin{aligned} & \mathbf{y} = W P_m^T \\ \Rightarrow \quad & P_m^T = W^{-1}\mathbf{y} \end{aligned} \tag{4.6}$$

where each row of $\mathbf{y}$ represents an independent image basis.

Finally, an ICA representation of an image is obtained as follows: The set of images in $\mathbf{x}$ can be represented by their coordinates in the basis of eigenvectors, $R_m = \mathbf{x}P_m$. A minimum squared error approximation of $\mathbf{x}$ is obtained by

$$\mathbf{x_{rec}} = R_m P_m^T = \mathbf{x} P_m P_m^T. \tag{4.7}$$

Substituting (4.6) into (4.7), we get

$$\mathbf{x_{rec}} = R_m W^{-1}\mathbf{y} = \mathbf{x} P_m W^{-1}\mathbf{y}. \tag{4.8}$$

The rows of $\mathbf{x}P_m W^{-1}$ are the coefficients for the linear combination of independent bases in $\mathbf{y}$. Thus for the representation of a test image, which is a row vector $I_{1\times N}$, the ICA representation is

$$\mathbf{c} = I P_m W^{-1} \tag{4.9}$$

where $P_m W^{-1}$ is obtained during the ICA training procedure.

Figures 4.6 and 4.7 show two sets of learned independent bases of two different dimensionalities.

Method 2: Independent Projection Coefficients

In Method 2, we view an image as an observation of a random vector, and reduce its dimension from N, the total pixel number, to m. Let $\mathbf{x}_m$ denote the matrix containing the dimension-reduced images in its columns. Let $\{p_i\}_{i=1}^m$ again denote the eigenvectors that correspond to the m maximal eigenvalues of the covariance matrix of $\mathbf{x}$, and let P_m denote the matrix whose columns are $\{p_i\}_{i=1}^m$. Then we have

$$\mathbf{x}_m = P_m^T \mathbf{x} \tag{4.10}$$

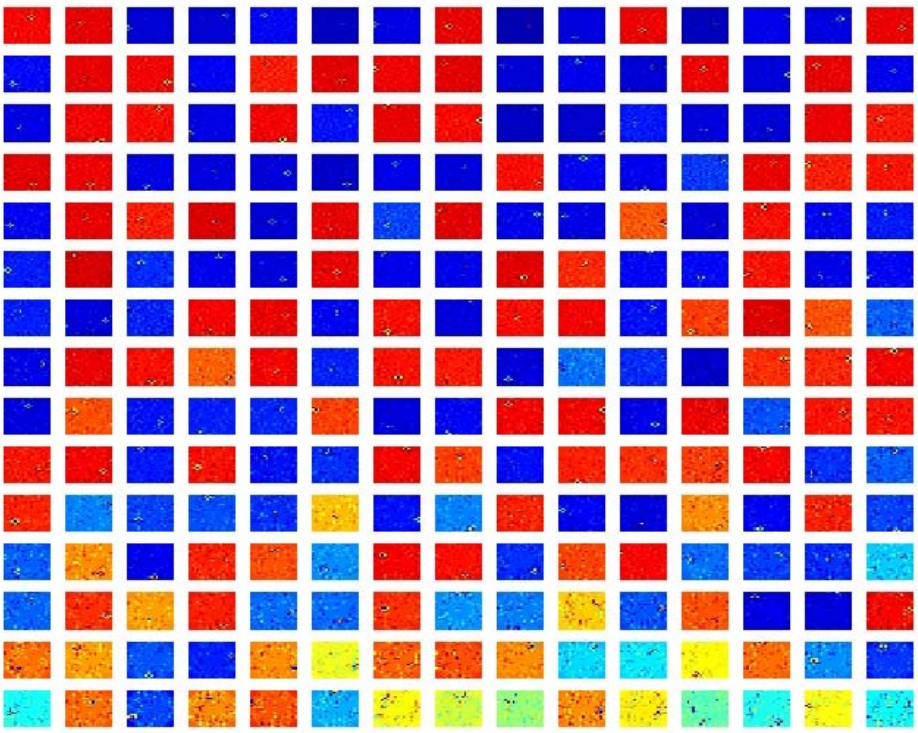

Figure 4.6: Independent basis images obtained from 230 eigenvectors

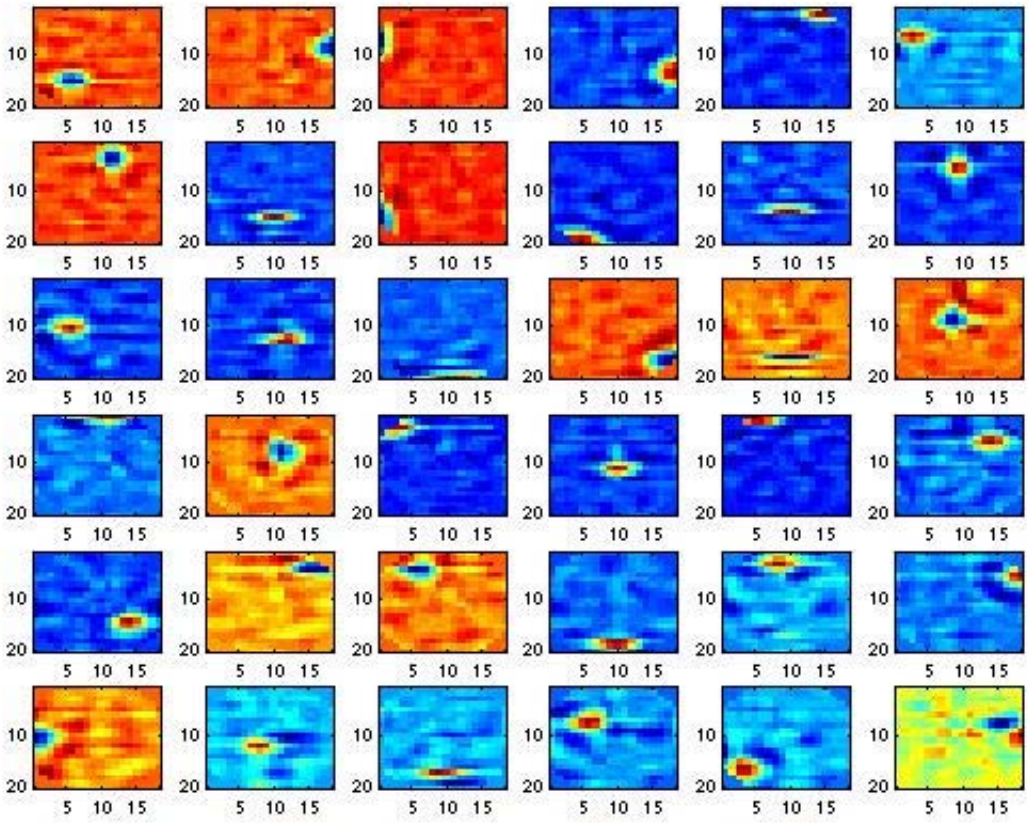

Figure 4.7: Independent basis images obtained from 45 eigenvectors

We apply the ICA algorithm to $\mathbf{x}_m$ and obtain

$$\mathbf{y} = W\mathbf{x}_m = WP_m^T\mathbf{x} \tag{4.11}$$

So for a test image which is represented by a column vector $I_{N\times 1}$, its ICA representation, $\mathbf{c}$, is given by

$$\mathbf{c} = WP_m^T I \tag{4.12}$$

where the matrix product WP_m^T is obtained in the training procedure. Denote the product WP_m^T by U. The columns of U are the basis images for the ICA representation in Method 2. Note that here every component in the representation vector $\mathbf{c}$ is independent of every other component, while in method 1, each basis image is independent of every other basis image. Details are provided in Bartlett et al. [8].

Figure 4.8 shows a set of learned basis images when the dimensionality, m, is 80.

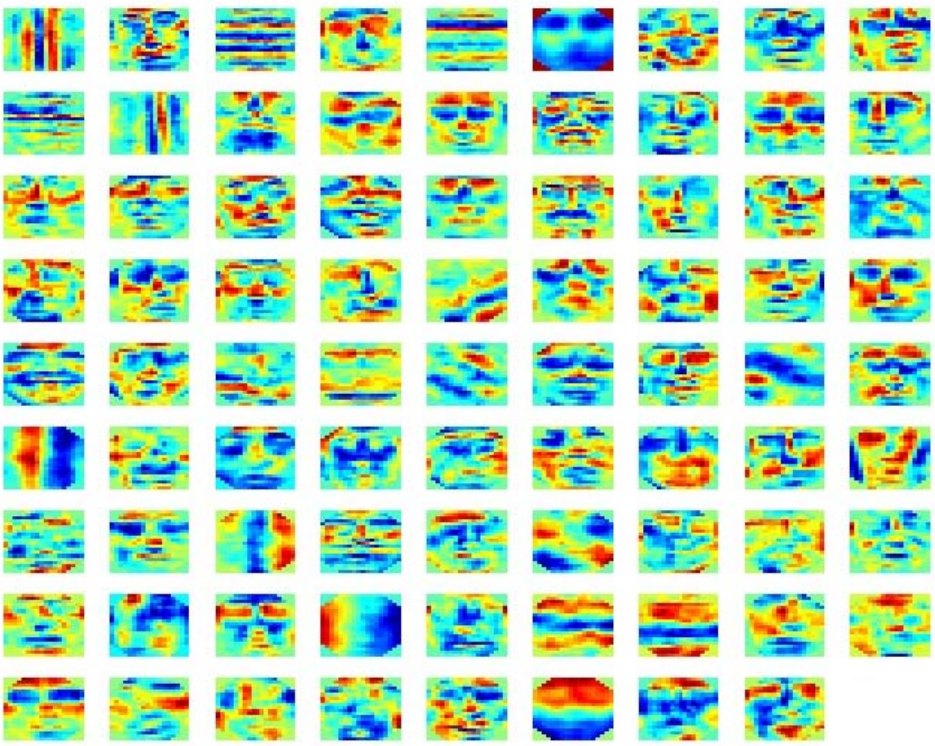

Figure 4.8: The ICA basis images obtained from 80 eigenvectors by Method 2

4.4.2 SVM Classification of ICA Features

After obtaining ICA features, we build the SVM training set $\{\mathbf{c}_i, d_i\}_{i=1}^{l}$ where d_i is the class type of feature $\mathbf{c}_i$. For the face detection problem, $d_i = 1$ when $\mathbf{c}_i$ is extracted from a face image, and $d_i = -1$ when $\mathbf{c}_i$ is from a non-face image. l is the size of the training set. SVM for

classification has been discussed in detail in Chapter 3. Here we mention another point about SVM. For a large training data set, the SVM training procedure is time consuming. How to speed up SVM training is an important research topic in the SVM research community. But different training strategies should only affect the training speed, not the learning ability of SVM. Here we used the method described in [49] for training.

4.5 The Hybrid ICA/SVM based Face Detection System

In this section we describe how to build a complete face detection system based on the hybrid ICA/SVM learning scheme. The detection system includes training and testing parts. The training part consists of the following steps:

1. In a training set for face detection, face and non-face patterns are assigned to 1 and -1 respectively. Each of these patterns has 20×20 pixels. The face patterns include faces with different facial expressions and under different views; see Figures 4.9 and 4.10 for some examples.

2. The data is preprocessed to compensate for variations in the training patterns:

   - In order to reduce background noise, pixels close to the boundary of each rectangular training pattern are removed by a binary mask.
   - Histogram equalization is then performed to compensate for illumination difference, etc.

3. After preprocessing, the ICA algorithm is applied to the data to learn the independent image bases which are used for feature extraction. Since two different ICA feature extraction methods can be applied, we can obtain two different set of image bases and features.

4. Using the ICA features, the SVM is trained to construct a decision plane in a high-dimensional space. Since it is difficult to find a good representative set of non-face patterns, a *bootstrapping* technique is used to add mis-classified non-face patterns into the training set, and then the SVM is re-trained to get a better decision plane.

Figures 4.9, 4.10, 4.11, and 4.12 show sets of image blocks after histogram equalization. They are used by ICA during the training procedure and include face patterns with different facial expressions and under slightly different views, non-face patterns used in initial training, and non-face patterns that are first misclassified and then used for bootstrapping.

The testing part comprises the following steps:

1. A skin color filter is used to find a binary mask which locates the face candidate regions in a test image.

2. The test image is rescaled several times, because we do not have prior knowledge about the face size.

3. A 20×20 window is moved over the face candidate regions to select image blocks for detection.

4. ICA features are extracted from the image blocks, using the pre-stored image bases which are obtained during the training procedure. Note that we have two different schemes for feature extraction using two different sets of image bases.

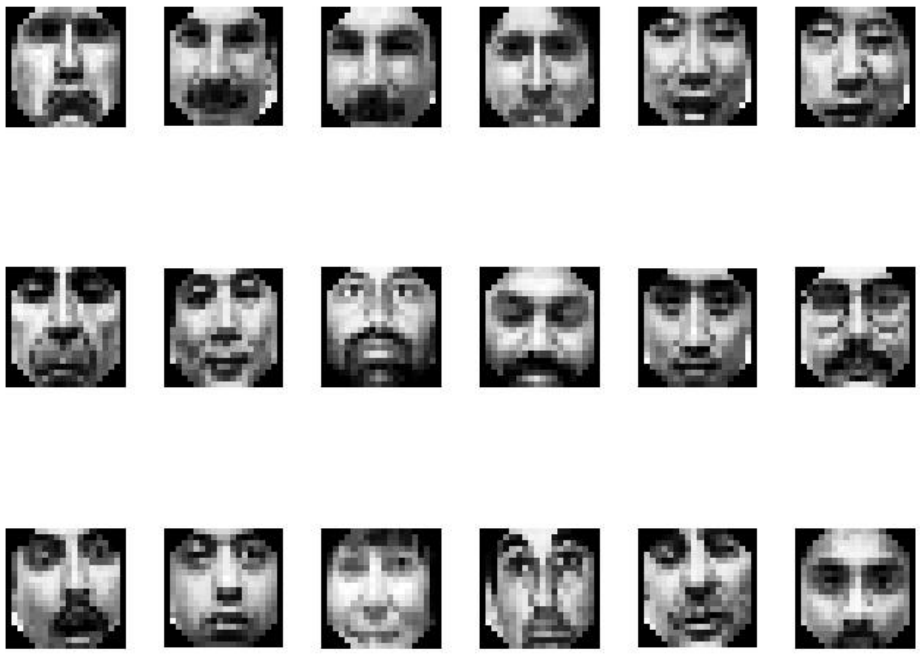

Figure 4.9: Face patterns with different facial expressions used in training

Figure 4.10: Face patterns under slightly different views used in training

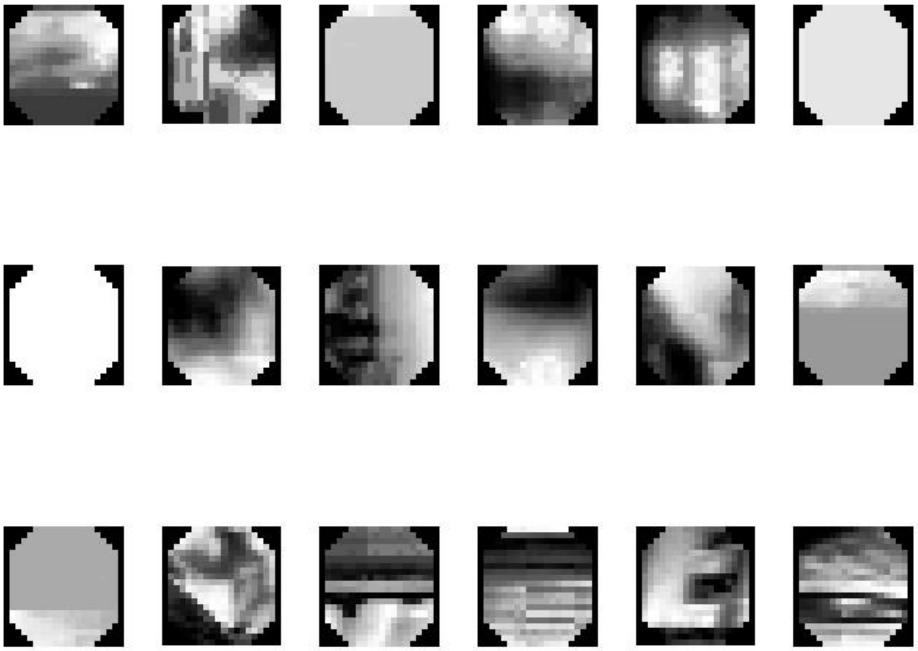

Figure 4.11: Non-face patterns used in initial training

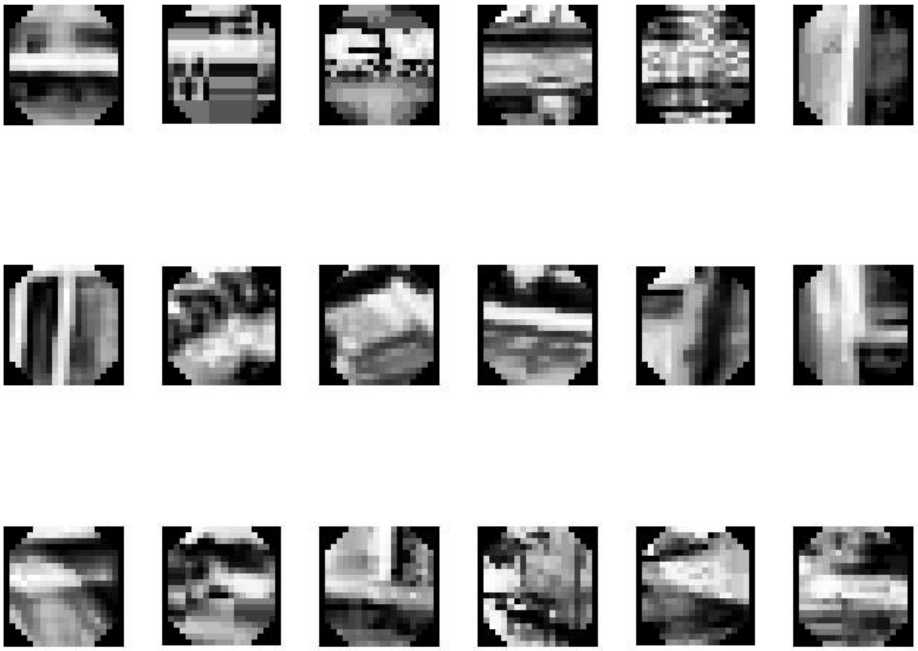

Figure 4.12: Non-face patterns used for bootstrapping

5. The trained SVM classifies the ICA features.

6. Post-processing is performed to enhance system performance:

   - If a detection appears at only one scale, it is usually a false detection. By ANDing the detection locations at different scales, we can effectively reduce the number of false detections.
   - The sliding window method usually leads to several detections near a face region. Thresholding the number of detections in a neighborhood tends to keep correct detections and eliminate false detections.
   - If a detection is correct, the detections that overlap the correct one are usually false. So after the previous two steps, the detection location with the largest number of detections within a neighborhood is assumed to be correct and preserved, while the other locations with fewer detections are eliminated.

7. The system takes the output of the post-processing as the final detection result.

4.6 Experimental Results

To evaluate the hybrid learning scheme on the face detection problem, we tested the system on 820 face examples from the LAMP face database developed by ourselves and from the Essex facial image database [23], as well as on 100884 nonface image blocks which we obtained from the LAMP face database and the web. In the LAMP face database, the face examples were recorded from TV shows. In the Essex facial image database, face examples have expression changes and position changes.

The results are reported in Table 4.1. From this table, we see that using ICA feature

Detection System	Number of Miss Detections	Number of False Detections
The Hybrid ICA/SVM Detection System based on ICA 1	39	54
The Hybrid ICA/SVM Detection System based on ICA 2	45	1743
The SVM Detection System without ICA Feature Extraction	41	252

Table 4.1: Face detection results

extraction Method 1, the hybrid learning scheme effectively improves the classification accuracy compared to the SVM detection system without ICA feature extraction. Several face detections on the test examples are shown in the following figures.

On the other hand, using ICA feature extraction Method 2 leads to deterioration of performance in classifying non-face examples. The possible reason might be the dimensionality of the features, which is 80 (reduced from 400) in our experiment, and may be too small to represent the original signals in Method 2.

Figure 4.13: Face detection example 1: at Scale 1

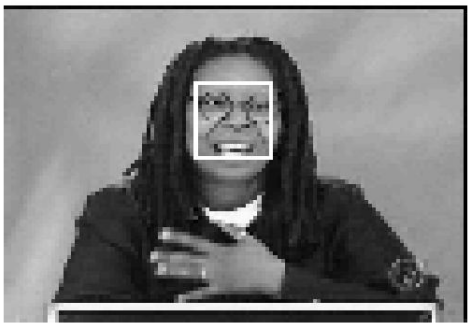

Figure 4.14: Face detection example 1: at Scale 2

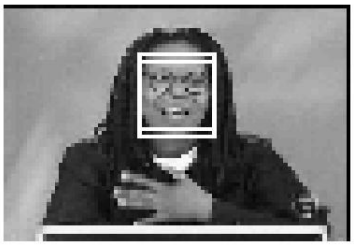

Figure 4.15: Face detection example 1: at Scale 3

Figure 4.16: Face detection example 2: Final result

Figure 4.17: Face detection example 3: Final result

Figure 4.18: Face detection example 4: Final result

4.7 Conclusion and Discussion

In this chapter, we have presented a new hybrid supervised/unsupervised learning scheme that integrates ICA and SVM to address pattern recognition problems.

In low-level feature extraction, ICA finds independent bases or coefficients to represent data. From Figures 4.6, 4.7 and 4.8, we see that the ICA bases emphasize edge information in the image data, as argued in [9]. In addition, because ICA tries to make bases or representative coefficients independent of each other, the ICA features represent the data better than the PCA features when the training data are not orthogonal to each other in the probability sense—for example, face image data with different facial expressions and in different views. In high-level feature classification, as an approximate implementation of the SRM Principle, the SVM tends to give good generalization performance. Many applications of SVM have proven this point. A common characteristic of ICA and SVM is sparsity. The ICA output is sparse. As shown in [48], ICA is formally equivalent to sparse coding under some condition. The support vectors whose linear combination comprises the trained SVM are also sparse. In [29], Girosi proves an equivalence between SVM and a Sparse Approximation (SA) scheme under noise-free condition. Thus combining ICA and SVM yields a hierarchical sparse learning scheme. Experimental results on the face detection problem show that the hybrid ICA/SCM learning scheme effectively improves detection system performance, compared with applying SVM directly to the original image data.

An idea that should improve this hybrid learning scheme is integrating SVM with subband-based ICA, which was proposed in Chapter 2. By applying ICA in the time-frequency plane, subband-based ICA successfully separates mixed acoustic signals, such as speech and music signals, even in the presence of strong noise or when performed on-line. Though here the aim is not to separate mixed acoustic signals, subband-based ICA is expected to find better signal representations, because it leads to a more sparse output than classical ICA algorithms and is more robust against noise. Naturally, subband-based ICA can be extended to multi-scale feature extraction. Since ICA bases are wavelet-like when trained on natural image data, an interesting similarity between subband-based ICA feature extraction and human auditory (or visual) processing is that both of them have a two-layered wavelet-like structure.

In addition, inspired by the use of the kernel trick in SVM, we hope to construct a kernel-based ICA algorithm. The idea is to apply ICA to the high-dimensional nonlinear mapped space instead of the original signal space. Kernel-based ICA is expected to have applications to more efficient feature extraction and separation of nonlinear mixed signals.

Finally, we would like to point out that the hybrid ICA/SVM scheme is a general learning scheme, which can be applied to other problems than face detection, such as speech processing and data mining.

Chapter 5

Conclusions

Machine learning algorithms play increasingly important roles in many areas, such as pattern recognition, signal processing, and communications. In this thesis, we have proposed two machine learning schemes, Subband-based Independent Component Analysis scheme and the hybrid Independent Component Analysis/Support Vector Machine scheme, and applied them to the problems of blind acoustic signal separation and face detection.

Inspired by our understanding of the subbanding strategies used in the early auditory system, we have proposed subband-based ICA, a new powerful learning algorithm, to solve the blind source separation (BSS) problem (Chapter 2). Though classical ICA algorithms have been applied to address the BSS problem, they do not work well in the presence of noise or when performed on-line. By performing separation in several frequency bands which contain most of the energy in the mixture, the new subband-based ICA approach is robust against noise and converges to the real demixing matrix quickly, even in its on-line version. The experimental results, as shown in Figures 2.3, 2.4, and 2.5, demonstrate its success while other ICA algorithms fail. The virtually increased signal-to-noise ratio in those frequency bands, the fact that subband signals, i.e., wavelet coefficients, are more peaky and heavy-tailed distributed than the original signals, and the adaptation to the properties of the signal and noise by the incorporation of a best basis selection algorithm, all contribute to the success of subband-based ICA.

Subband-based ICA is also a computationally efficient algorithm because it reduces computational complexity by performing separation in the down-sampled signals in several or even a single frequency band. Its speed is much higher than those of previous ICA algorithms, as shown in Tables 2.2 and 2.3.

We can further generalize subband-based ICA by replacing the subband decomposition with some appropriate projection. For example, a nonlinear projection can be used under some criterion, e.g., maximum likelihood, to derive a nonlinear ICA.

Our future work on the blind separation problem will include using some signal cues, for example, the pitches of acoustic signals, and available prior knowledge to guide separation. In this way, we may increase convergence speed and accomplish the separation even in cases where the number of sensors is less than the number of sources. Some work has been initiated in this direction.

Subband based ICA is, in essence, an unsupervised learning scheme. In Chapter 3, a supervised learning algorithm, the Support Vector Machine (SVM), is presented. As an approximate implementation of the Structural Risk Minimization (SRM) Principle that is proposed in statistical learning theory, SVM provides a method of minimizing the sum of the number of training errors and the VC dimension, which indicates the model complexity, so that high generalization performance can be achieved.

In addition to high generalization performance, SVM can control model complexity independently of the dimensionality of the feature space, by reformulating the primal optimization problem into its dual problem and using an inner-product kernel trick. Actually, an infinite feature space is allowed in SVM. Moreover, the convex cost function in the QP problem guarantees that SVM will find a globally optimal solution that automatically tunes all the parameters in the learning scheme, while many other learning algorithms suffer from falling into local extrema.

Though originally derived from the SRM Principle to address the problem of the tradeoff between model complexity and generalization ability, SVM is closely related to other known techniques and research problems:

- The support vectors are usually sparse. They only constitute a fraction of the total number of examples in the training set. Using the reproducing property of the Reproducing Kernel Hilbert Space (RKHS), Girosi [29] shows an equivalence between SVMs in the noiseless case and a Sparse Approximation (SA) scheme that resembles the Basis Pursuit De-Noising algorithm [14].

- Also in [29], Girosi gives a derivation of the SVM algorithm in the framework of regularization theory. In [24], Evgeniou et al. give a unified framework for regularization networks and SVM. The reformulation of SVM in regularization theory reveals the connection between SVM and other known techniques. However, it hides the relation between SVM and the SRM Principle.

SVM provides high generalization performance without incorporating any prior knowledge about the problem. An important research topic is how to incorporate problem-domain knowledge into SVM to further improve its performance. Some proposed approaches include adding an additional term that represents prior knowledge in the cost function, using prior knowledge to design the kernel function [59], and adding virtual examples into the training set [58]. More efficient and natural ways of adding prior knowledge into SVM are yet to be developed. For example, integrating Bayesian learning theory into SVM might be a good way of exploiting prior information.

Another research topic related to SVM is that the kernel trick in SVM can also be used in other algorithms that are based on the inner product of the data. For example, Principal Component Analysis can be done in a high-dimensional feature space by using a suitable nonlinear kernel function [60]. Fisher discriminant analysis also uses a similar idea [45].

Finally in Chapter 4, we have presented a new hybrid supervised/unsupervised learning scheme that integrates ICA and SVM to address pattern recognition problems.

In low-level feature extraction, ICA finds independent bases or coefficients to represent data. From Figures 4.6, 4.7 and 4.8, we can see that the ICA bases emphasize edge information in the image data, as argued in [9]. In addition, because ICA tries to make data bases or representation coefficients independent of each other, the ICA features represent the data better than the PCA features when the training data are not orthogonal to each other in a probability sense—for example, face image data with different facial expressions and seen in different views. In high-level feature classification, as an approximate implementation of the SRM Principle, SVM tends to have good generalization performance. Many applications of SVM have proven this point. One common characteristic shared by ICA and SVM is sparseness. The ICA output is sparse. As shown in [48], ICA is formally equivalent to sparse coding under some conditions. The support vectors whose linear combination comprises the trained SVM are also sparse. Thus combining ICA and SVM yields a hierarchical sparse learning scheme. Experimental results on

the face detection problem show that the hybrid ICA/SCM learning scheme effectively improves detection system performance, compared with applying SVM directly to the original image data.

One idea that should improve this hybrid learning scheme is integrating SVM with subband-based ICA, which was proposed in Chapter 2. By applying ICA in the time-frequency plane, subband-based ICA successfully separates mixed acoustic signals, such as speech and music signals, even in the presence of strong noise or when performed on-line. Though here the aim is not to separate mixed acoustic signals, subband-based ICA is expected to find better signal representations, because it leads to a more sparse output than classical ICA algorithms and is more robust against noise. Naturally, subband-based ICA can be extended to multi-scale feature extraction. Since ICA bases are wavelet-like when trained on natural image data, an interesting similarity between subband-based ICA feature extraction and human auditory (or visual) processing is that both of them have a two-layered wavelet-like structure.

In addition, inspired by the use of the kernel trick in SVM, we hope to construct a kernel-based ICA algorithm. The idea is to apply ICA to the high-dimensional nonlinear mapped space instead of the original signal space. Kernel-based ICA is expected to have applications to more efficient feature extraction and separation of nonlinear mixed signals.

Finally, we would like to point out that the hybrid ICA/SVM learning scheme is a general scheme, which can be applied to other problems than face detection, for example, speech processing or data mining.

Bibliography

[1] J. B. Allen. How do humans process and recognize speech? *IEEE Trans. on Speech and Audio Processing*, 2:567–577, 1994.

[2] S. Amari, T. P. Chen, and A. Cichocki. Non-holonomic orthogonal learning algorithm for blind source separation.

[3] S. Amari and A. Cichocki. Adaptive blind signal processing—neural network approaches. *Proceedings of the IEEE*, 86:2026–48, 1998.

[4] S. Amari, A. Cichocki, and H. H. Yang. A new learning algorithm for blind signal separation. In D. Touretzky, M. Mozer, and M. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, pages 752–763. MIT Press, Cambridge, MA, 1996.

[5] J. J. Atick. Entropy minimization: A design principle for sensory perception? *International Journal of Neural Systems*, 3:81–90, 1992.

[6] H. B. Barlow. Unsupervised learning. *Neural Computation*, 1:295–311, 1989.

[7] H. B. Barlow. What is the computational goal of the neocortex ? In *Large-scale Neuronal Theories of the Brain*. MIT Press, Cambridge, MA, 1994.

[8] M. S. Bartlett, H. M. Lades, and T. J. Sejnowski. Independent component representations for face recognition. In *Proc. SPIE Conf. on Human Vision and Electronic Imaging III*, volume 3299, pages 528–539, 1998.

[9] A. J. Bell and T. J. Sejnowski. Edges are the independent components of natural scenes. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, page 831. MIT Press, Cambridge, MA, 1997.

[10] V. Blanz, B. Schölkopf, H. Bülthoff, C. Burges, V. Vapnik, and T. Vetter. Comparison of view-based object recognition algorithms using realistic 3D models. In C. von der Malsburg, W. von Seelen, J. C. Vorbrüggen, and B. Sendhoff, editors, *Artificial Neural Networks—ICANN'96*, pages 251–256, Berlin, 1996. Springer Lecture Notes in Computer Science, Vol. 1112.

[11] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Knowledge Discovery and Data Mining*, 2(2), 1998.

[12] J.-F. Cardoso. Blind signal separation: statistical principles. *Proceedings of the IEEE*, 86:2009–2025, 1998.

[13] R. Chellappa, C. L. Wilson, S. Sirohey, and C. S. Barnes. Human and machine recognition of faces: A survey. Technical Report CS-TR-3339, Department of Computer Science, University of Maryland, College Park, August 1994.

[14] S. S. B. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. Technical Report 409, Department of Statistics, Stanford University, 1995.

[15] R. Coifman and M. V. Wickerhauser. Entropy-based algorithms for best basis selection. *IEEE Trans. on Information Theory*, 38:713–719, 1992.

[16] A. J. Colmenarez and T. S. Huang. Face detection with information based maximum discrimination. In *Proc. CVPR*, pages 782–787, 1997.

[17] T. M. Cover. Geometrical and statistical properties of systems and linear inequalities with applications in pattern recognition. *IEEE Trans. on Electronic Computers*, 19:326–334, 1965.

[18] Y. Dai, Y. Nakano, and H. Miyao. Extraction of facial images from a complex background using SGLD matrices. In *Proc. ICPR*, pages A:137–141, 1994.

[19] I. Daubechies. *Ten Lectures on Wavelets*. SIAM, 1992.

[20] G. Deco and D. Obradovic. Linear redundancy reduction learning. *Neural Networks*, 8:751–755, 95.

[21] G. L. Donato, M. S. Bartlett, J. C. Hager, P. Ekman, and T. J. Sejnowski. Classifying facial actions. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21:974–989, 1999.

[22] D. L. Donoho. De-noising by soft-thresholding. *IEEE Trans. on Information Theory*, 41:613–627, 1995.

[23] The Essex facial image database, http://cswww.essex.ac.uk/mv/projects.html.

[24] T. Evgeniou, M. Pontil, and T. Poggio. A unified framework for regularization networks and support vector machines. Technical Memo AIM-1654, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, December 1999.

[25] D. J. Field. What is the goal of sensory coding? *Neural Computation*, 6:559–601, 1994.

[26] M. M. Fleck, D. A. Forsyth, and C. Bregler. Finding naked people. In *Proc. 4th European Conference on Computer Vision*, 1996.

[27] R. Fletcher. *Practical Methods of Optimization,* 2nd ed. Wiley, New York, 1987.

[28] Z. Ghahramani and G. E. Hinton. The EM algorithm for mixtures of factor analyzers. Technical Report CRG-TR-96-1, University of Toronto, 1996.

[29] F. Girosi. An equivalence between sparse approximation and support vector machines. *Neural Computation*, 10:1455–1480, 1998.

[30] H. P. Graf, E. Cosatto, D. Gibbon, M. Kocheisen, and E. Petajan. Multimodal systmem for locating heads and faces. In *Proc. IEEE Int'l. Conf. on Automatic Face and Gesture Recognition*, pages 88–93, 1996.

[31] S. Haykin. *Neural networks: A Comprehensive Foundation.* Prentice Hall, Upper Saddle River, NJ, 2nd edition, 1999.

[32] P. J. Huber. Projection pursuit. *Annals of Statistics*, 13:435–475, 1985.

[33] http://www.cis.hut.fi/projects/ica/fastica/.

[34] A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans. on Neural Networks*, 10:626–634, 1999.

[35] A. Hyvärinen. Survey on independent component analysis. *Neural Computing Surveys*, 2:94–128, 1999.

[36] A. Hyvärinen, P. Hoyer, and E. Oja. Sparse code shrinkage: Denoising by nonlinear maximum likelihood estimation. In *Advances in Neural Information Processing Systems*, volume 11, 1999.

[37] A. Hyvärinen and P. Pajunen. Nonlinear independent component analysis: Existence and uniqueness results. *Neural Networks*, 12:429–439, 1999.

[38] http://sound.media.mit.edu/ica-bench/sources/.

[39] T.-W. Lee, M. Girolami, and T. J. Sejnowski. Independent component analysis using an extended infomax algorithm for mixed sub-Gaussian and super-Gaussian sources. *Neural Computation*, 11:409–433, 1999.

[40] T.-W. Lee, B. U. Koehler, and R. Orglmeister. Blind source separation of nonlinear mixing models. In *Proc. IEEE Int'l. Workshop on Neural Networks for Signal Processing*, pages 406–415, 1997.

[41] http://www.cnl.salk.edu/ tewon/.

[42] C. Liu and H. Wechsler. Comparative assessment of independent component analysis (ICA) for face recognition. In *Proc. Int'l. Conf. on Audio- and Video-based Biometric Person Authentication*, 1999.

[43] R. Lyon and S. Shamma. Auditory representations of timbre and pitch. In *Auditory Computation*, pages 221–270. Springer, Berlin, 1995.

[44] J. Mercer. Functions of positive and negative type, and their connection with the theory of integral equations. *Transactions of the London Philosophical Society (A)*, 209:415–446, 1909.

[45] S. Mika, G. Rätsch, J. Weston, B. Schölkopf, and K.-R. Müller. Fisher discriminant analysis with kernels. In Y.-H. Hu, J. Larsen, E. Wilson, and S. Douglas, editors, *Neural Networks for Signal Processing IX*, pages 41–48. IEEE, 1999.

[46] B. Moghaddam and A. P. Pentland. Probabilistic visual learning for object recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19:696–710, 1997.

[47] B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996.

[48] B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision Research*, 37:3311–3325, 1997.

[49] E. Osuna, R. Freund, and F. Girosi. Training support vector machines: An application to face detection. In *Proc. CVPR*, pages 130–136, 1997.

[50] R. J. Qian and T. S. Huang. Object detection using hierarchical MRF and MAP estimation. In *Proc. CVPR*, pages 186–192, 1997.

[51] A. N. Rajagopalan, K. S. Kumar, J. Karlekar, R. Manivasakan, M. M. Patil, U. B. Desai, P. G. Poonacha, and S. Chaudhuri. Finding faces in photographs. In *Proc. ICCV98*, pages 640–645, 1998.

[52] R. A. Redner and H. F. Walker. Mixture densities, maximum likelihood and the EM algorithm. *SIAM Review*, 26:195–239, 1984.

[53] D. Roth, M. Yang, and N. Ahuja. A SNoW-based face detector, 2000.

[54] H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20:23–38, 1998.

[55] H. A. Rowley, S. Baluja, and T. Kanade. Rotation invariant neural network-based face detection. In *Proc. CVPR*, page 963, 1998.

[56] F. S. Samaria. *Face Detection Using Hidden Markov Models.* PhD thesis, University of Cambridge, 1994.

[57] M. Schmidt. Identifying speaker with support vector machine. In *Interface'96*, Sydney, 1996.

[58] B. Schölkopf, C. Burges, and V. Vapnik. Incorporating invariances in support vector learning machines. In C. von der Malsburg, W. von Seelen, J. C. Vorbrüggen, and B. Sendhoff, editors, *Artificial Neural Networks*, volume 1112 of *Lecture Notes in Computer Science*, pages 47–52. Springer, Berlin, 1996.

[59] B. Schölkopf, P. Y. Simard, A. J. Smola, and V. N. Vapnik. Prior knowledge in support vector kernels. In M. I. Jordan, M. J. Kearns, and S. A. Solla, editors, *Advances in Neural Information Processing Systems*, volume 10, pages 640–646. MIT Press, Cambridge, MA, 1998.

[60] B. Schölkopf, A. Smola, and K.-R. Müller. Kernel principal component analysis. In B. Schölkopf, C. J. C. Burges, and A. J. Smola, editors, *Advances in Kernel Methods—SV Learning*, pages 327–352. MIT Press, Cambridge, MA, 1999.

[61] E. P. Simoncelli and E. H. Adelson. Noise removel via Bayesian wavelet coring. In *IEEE Int'l. Conf. on Image Processing*, 1996.

[62] K. K. Sung and T. Poggio. Example-based learning for view-based human face detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20:39–51, 1998.

[63] J. H. van Hateren and A. van der Schaaf. Independent component filters of natural images compared with simple cells in primary visual cortex. *Proc. Royal Society*, 265 B:359–366, 1998.

[64] V. Vapnik. *Estimation of Dependences Based on Empirical Data* (in Russian). Nauka, Moscow, 1979. (English translation: Springer, New York, 1982).

[65] V. Vapnik. *The Nature of Statistical Learning Theory.* Springer, New York, 1995.

[66] J. Yang and A. Waibel. A real-time face tracker. In *Proc. WACV*, 1996.

[67] M. H. Yang, N. Ahuja, and D. Kriegma. A survey on face detection methods, 1999.

[68] X. Yang, K. Wang, and S. Shamma. Auditory representations of acoustic signals. *IEEE Trans. on Information Theory*, 38:824–839, 1992.

[69] A. L. Yuille, D. S. Cohen, and P. W. Hallinan. Feature extraction from faces using deformable templates. *IJCV*, 8:99–111, 1992.